NASA Contractor Report 179429

Application of a Sensitivity Analysis Technique to High-Order Digital Flight Control Systems

James D. Paduano and David R. Downing

(NASA-CR-179429) APPLICATION OF A SENSITIVITY ANALYSIS TECHNIQUE TO HIGH-ORDER DIGITAL FLIGHT CONTROL SYSTEMS (Kansas Univ. Center for Research) 155 p Avail: NTIS HC A09/MF A01 CSCL 01C G3/08

N87-28565

Unclas 0098639

Contract NCC 2-293
September 1987

National Aeronautics and Space Administration

NASA Contractor Report 179429

Application of a Sensitivity Analysis Technique to High-Order Digital Flight Control Systems

James D. Paduano and David R. Downing
Flight Research Laboratory, University of Kansas Center for Research, Inc., Lawrence, Kansas 66045

Prepared for
Ames Research Center
Dryden Flight Research Facility
Edwards, California
Under Contract NCC 2-293

1987

National Aeronautics and
Space Administration
Ames Research Center
Dryden Flight Research Facility
Edwards, California 93523-5000

"APPLICATION OF A SENSITIVITY ANALYSIS TECHNIQUE TO HIGH-ORDER DIGITAL FLIGHT CONTROL SYSTEMS"

Abstract

A sensitivity analysis technique for multiloop flight control systems is studied. This technique uses the scaled singular values of the return difference matrix as a measure of the relative stability of a control system. It then uses the gradients of these singular values with respect to system and controller parameters to judge sensitivity.

The sensitivity analysis technique is first reviewed; then it is extended to include digital systems, through the derivation of new singular-value gradient equations. These digital-system gradient equations are a necessary extension to the technique when real-world systems are to be analyzed. Gradients with respect to parameters which do not appear explicitly as control-system matrix elements are also derived, so that high-order systems can be studied.

A complete review of the integrated technique is given by way of a simple example: the inverted pendulum problem. The technique is then demonstrated on the X-29 control laws. The X-29 control system represents a high-order, digital, multiloop system and, as such, is a good test case for the technique. Results show that

linear models of real systems can indeed be analyzed by this sensitivity technique, if it is applied with care.

A computer program called SVA has been written to accomplish the singular-value sensitivity analysis technique. Thus computational methods and considerations form an integral part of many of the discussions in this paper. A user's guide to the program is included in an appendix. SVA is a fully public domain program, running on the NASA/Dryden Elxsi computer.

ACKNOWLEDGEMENTS

I would like to thank several people, both at the University of Kansas and at NASA Ames/Dryden, for their help and support. Dr. David Downing deserves special thanks for his encouragement and advice throughout this project and my graduate career. I would also like to thank the other members of my advisory committee--Dr. John Bunce, Dr. Edward Lan, and Dr. Jan Roskam--for their comments and suggestions. Alfredo Herrera also deserves mention, as this project's initiator and a very fine person to work with. At Dryden, I would like to thank Bob Clarke (some of the ideas in this thesis are his), Marty Brenner, and Trindel Maine, who all provided provocative discussion and unfailing interest. Their support was greatly appreciated. Finally, I would like to thank Nancy Hanson for her patience and accuracy in transcribing the many equations and corrections in this document.

TABLE OF CONTENTS

Page

Abstract....i
Acknowledgements....iii
List of Figures....vii
List of Tables....xi
List of Symbols and Acronyms....xii

1. Introduction....1

2. Background....8

2.1 The Need for a Multiloop Analysis Method....8

2.2 Vector and Matrix Norms, Singular Values, and Their Application....13

2.3 Elaboration of the Characterstics of Singular Values....26

2.4 Singular-Value Gradients....33

3. Computing Gradients for Digital Control Systems....45

3.1 The Effect of Discretization on the SVA....45

3.2 Derivation of the Singular-Value Gradients for Digital Systems....47
3.2.1 Digital Gradients for the Input Case....48
3.2.2 Digital Gradients for the Output Case....54

4. Application of the SVA to a Representative Example....58

4.1 An Inverted Pendulum with a Digital Controller....58

4.2 Step-by-Step Application of the SVA....68
4.2.1 Setting up the Matrices....68
4.2.2 Computations at Each Frequency....71
4.2.3 Data Reduction of the $\underline{\sigma}$-Gradient Plots....74

4.3 Application of the SVA to the Pendulum Example....76
4.3.1 $\underline{\sigma}$-Plots for the Pendulum Example....76
4.3.2 $\underline{\sigma}$-Gradient Plots for the Pendulum Example....81

TABLE OF CONTENTS, continued

Page

5. Application of the SVA to a Real System.....86

5.1 σ-Plots Applied to a Real System.....86
5.1.1 X-29 Longitudinal Mode.....87
5.1.2 X-29 Lateral Mode.....94
5.2 Singular-Value Gradients Applied to a Real System.....101
5.2.1 Aerodynamic Gradients.....102
5.2.2 Control System Parameter Gradients.....108

6. Conclusions and Recommendations.....113

References.....116

Appendix A: Derivations Concerning Singular Values.....A.1

A.1 Proofs for Section 2.....A.1

A.2 Proofs for Section 3.....A.6

A.3 Proofs for Section 4.....A.7

Appendix B: Description of the X-29A Advanced Technology Demonstrator.....B.1

Appendix C: User's Guide to SVA.....C.1

C.1 Basic Definitions.....C.2
C.1.1 Control System Description.....C.2
C.1.2 Gradients.....C.3

C.2 Using a CONTROL Input File to Run SVA.....C.7
C.2.1 Discussion.....C.7
C.2.1.1 Restrictions.....C.8
C.2.1.2 Partial-Sum Gradients.....C.11
C.2.2 Definitions.....C.12
C.2.3 Input Format.....C.13
C.2.4 To Run.....C.14

C.3 Input to SVA Using PRESVA.....C.17
C.3.1 Discussion.....C.17
C.3.2 Definitions.....C.17
C.3.3 Input Format.....C.18
C.3.4 To Run.....C.18

TABLE OF CONTENTS, continued

		Page
C.4	Data File Input to SVA	C.19
	C.4.1 Discussion	C.19
	C.4.2 Definitions	C.21
	C.4.2.1 Definitions for Partial-Sum Gradients	C.22
	C.4.3 Input Format	C.23
	C.4.4 To Run	C.26
C.5	Calling SVA as a Subroutine	C.29
	C.5.1 Calling Sequence	C.30
	C.5.2 Input Arguments	C.30
	C.5.3 Output Arguments	C.34
	C.5.4 COMMON Blocks	C.35
	C.5.5 Error Messages	C.35
	C.5.6 Subroutines Employed by SVANAL	C.36
	C.5.7 Subroutines Employing SVANAL	C.36

LIST OF FIGURES

Number	Title	Page
1.1	Block diagram of a basic feedback control system	2
2.1	Block diagram of a single-input single-output control system with a gain and phase perturbation at the input	9
2.2	Example of a Bode plot with misleading gain and phase margins	9
2.3	Diagram of classical technique for studying perturbations in multi-input multi-output systems	12
2.4	2-dimensional illustration of a matrix as a vector transformation	12
2.5	Block diagram of a multi-input multi-output control system with a perturbation at the input	16
2.6	Universal diagram for multiloop gain-phase margin evalutation (Reference [3])	20
2.7	Illustration of the minimum singular value of the return difference matrix on a Nyquist plot (SISO)	22
2.8	Example of a Nyquist plot with the same minimum singular value	22
2.9	Singular-value plot for the system whose Nyquist plot appears in Figure (2.7)	23
2.10	Locations for the disturbance matrix, L: 1) input node, 2) output node	32
2.11	Effect of a parameter variation on the $\underline{\sigma}$-plot, as predicted by the $\underline{\sigma}$-gradient	41
2.12	Possible flow chart of singular value sensitivity analysis technique	43
3.1	Allowable control system structure for singular value gradients of digital systems	51

LIST OF FIGURES, continued

Number	Title	Page
4.1	"An inverted pendulum positioning system" [16]	59
4.2	"Inverted pendulum: forces and displacements" [16]	59
4.3	Proportional-plus-integral control system (with digital lead-lag) for the inverted pendulum example of Kwakernaak and Sivan [16]	61
4.4	Results of the singular value analysis when applied to the pendulum example	73
4.5	Plot of 1) minimum singular value and 2) minimum eigenvalue for the pendulum example (unperturbed control system); input node	77
4.6	Plot of 1) minimum singular value and 2) minimum eigenvalue for the pendulum example (unperturbed control system); output node	78
4.7	$\underline{\sigma}$-plots at the input node of the pendulum example for various scalings. $\underline{\lambda}$-plot is included to show that it is an upper bound	80
4.8	$\underline{\sigma}$-plot/$\underline{\lambda}$-plot at the output node for the pendulum example with scaling	82
4.9	Plot of the singular value gradients with respect to 1) proportional gain (K1) and 2) compensator zero location ($-\rho_z$) for the pendulum example (unperturbed control system) input node	83
4.10	Plot of the singular value gradients with respect to 1) pendulum characteristic length (L') and 2) servo bandwidth (BW) for the pendulum example (unperturbed control system) input node	84
5.1	Simplified block diagram of the X-29 longitudinal primary flight control system	88
5.2	Singular value plot for the X-29 longitudinal mode at M=.9 and H=15,000 ft, with the loop broken at point A in Figure 5.1	89

LIST OF FIGURES, continued

Number	Title	Page
5.3	Singular value/eigenvalue plot for the X-29 longitudinal mode at M=.9 H=15,000 ft, with the loop broken at point B in Figure 5.1	91
5.4	Frequency response for the X-29 longitudinal control system (loop broken at point A in Figure 5.1)	92
5.5	Simplified block diagram of the X-29 lateral-directional primary flight control system	95
5.6	Singular value/eigenvalue plot for the X-29 lateral mode at M=.9 and H=15,000 ft	96
5.7	Aileron loop frequency response for the X-29 lateral-directional control system (loop broken at ZOH, rudder loop closed)	97
5.8	Rudder loop frequency response for the X-29 lateral-directional control system (loop broken at ZOH, aileron loop closed)	98
5.9	Plot of 1) singular value, 2) $\partial\underline{\sigma}/\partial M_{DC}$, and 3) $\partial\underline{\sigma}/M_{\alpha}'$ for the X-29 longitudinal mode at M=.9 and H=15,000 ft	103
5.10	Plot of percent variation in M_{α}' required to drive $\underline{\sigma}$-plot minimum to .2 for the X-29 longitudinal mode at M=.9, H=15,000 ft	106
5.11	Plot of percent variation in M_{DC} required to drive $\underline{\sigma}$-plot to .2 for the X-29 longitudinal mode at M=.9 and H=15,000 ft	107
5.12	Effect of 1) $\partial\underline{\sigma}/\partial$GXG2 on 2) $\underline{\sigma}$ for the X-29 at M=.9 and H=15,000 ft. 3) is the plot which results when GXG2 is perturbed in the negative direction by 50%	112

LIST OF FIGURES, continued

Number	Title	Page
B.1	X-29A Advanced Technology Demonstrator	B.2
C.1	Structure of control system for input to CONSVA	C.9

LIST OF TABLES

Number	Title	Page
4.1	Parameter sensitivities for the pendulum example at the input node: variations necessary to drive $\underline{\sigma}_{min}$ to .06 (half the original value)	85
4.2	Parameter sensitivities for the pendulum example at the output node: variations necessary to drive $\underline{\sigma}_{min}$ to .06 (half the original value)	85
5.1	Gain and phase margin comparisons for the longitudinal X-29 at M=.9, H=15,000 ft	93
5.2	Gain and phase margin comparisons for the lateral X-29 at M=.9, H=15,000 ft	99
5.3	Sensitivity of the X-29 to changes in aerodynamic and control power derivatives, both longitudinal and lateral. Parameters are in decending order of importance (M=.9, H=15,000 ft)	104
5.4	Sensitivity of the X-29 to changes in longitudinal control system parameters. Parameters are in decending order of importance (M=.9, H=15,000 ft)	109
5.5	Sensitivity of the X-29 to changes in lateral control system parameters. Parameters are in decending order of importance (M=.9, H=15,000 ft)	110
B.1	Longitudinal dynamic and control-power matrices for the X-29 at M= .9 and H=15,000 ft	B.3
B.2	Gains and transfer function coefficients for the X-29 longitudinal control system - Figure 5.1 (M=.9, H=15,000 ft)	B.5
B.3	Lateral-directional dynamic and control-power matrices for the X-29 at M=.9 and H=15,000 ft	B.6
B.4	Gains and transfer function coefficients for the X-29 lateral-directional control system - Figure 5.5 (M=.9, H=15,000 ft)	B.7

LIST OF SYMBOLS AND ACRONYMS

Symbol	Definition
a_{ij}	Element of the $\bar{A}$ matrix: i-th row, j-th column
a_c	Element of the A_c matrix
A, B	Time-domain system dynamic matrices
$\bar{A}$, $\bar{B}$	Time-domain augmented-system dynamic matrices
A_1	Ratio of I_{xz}/I_{xx}
A_a, B_a	Dynamics matrices of the controller (augmentation)
A_c, B_c	Continuous dynamic matrices in a sampled-data system
A_d, B_d	Digital dynamic matrices in a sampled-data system
A_p, B_p	Plant dynamic matrices
b_{ij}	Element of the $\bar{B}$ matrix: i-th row, j-th column
b_c	Element of the B_c matrix
B_1	Ratio of I_{xz}/I_{zz}
c_{ij}	Element of the $\bar{C}$ matrix: i-th row, j-th column
C	Feedback matrix (also subscript for canard)
$\bar{C}$	Augmented feedback matrix
C_1,C_2	Interconnection matrices between the plant and controller
d	Pendulum displacement
dB	Decibels (x in decibels is $20 \cdot \log_{10}(x)$)
D'	Scaling matrix
DC	delta-canard (X-29)
DDF	delta-differential flap (X-29)
DF	differential flap (X-29)

Symbol	Definition
DR	delta-rudder (X-29)
DSF	delta-symmetric flap (X-29)
DST	delta-strake (X-29)
F	Friction force in pendulum example
g	Plant transfer function (s-domain, SISO)
G	Plant transfer function matrix (s-domain)
$G_{2:}$	Digital domain augmentation to the discrete control power matrix
h, h_1, h_2	Controller transfer functions (s-domain, SISO)
H	Controller transfer function matrix (s-domain); also altitude
H_a, F_a	Output equation matrices for the controller (augmentation)
H_d, F_d	Output equation matrices for the digital controller
H_p, F_p	Output equation matrices for the plant
i	Counter, also represents row position of matrix element
I	Identity matrix
j	Imaginary term $\sqrt{-1}$, also column position for matrix element
J	Matrix which destabilizes the control system; also pendulum moment of inertia
k	Digital domain counter, also general gain perturbation
K_1	Proportional gain in pendulum example
K_2	Integeral gain in pendulum example
l	SISO disturbance term, $l=ke^{j\phi}$
L	MIMO diagonal disturbance matrix, also pendulum length

Symbol	Definition
L'	Pendulum characteristic length
L_{12},	Digital domain augmentation to the transition matrix
L_{21}, L_{11}	(indices indicate row, column position)
L_p	Dimensional variation of rolling moment with roll rate
L_p'	A-matrix L_p derivative - $L_p' = (L_p + A_1 N_p)/(1 - A_1 B_1)$
L_r	Dimensional variation of rolling moment with yaw rate
L_r'	A-matrix L_r derivative - $L_r' = (L_r + A_1 N_r)/(1 - A_1 B_1)$
L_β	Dimensional variation of rolling moment with sideslip
L_β'	A-matrix L_β derivative - $L_\beta' = (L_\beta + A_1 N_\beta)/(1 - A_1 B_1)$
L_{DR}	Dimensional variaton of rolling moment with rudder
L_{DR}'	B-matrix L_{DR} derivative - $L_{DR}' = (L_{DR} + A_1 N_{DR})/(1 - A_1 B_1)$
L_{DDF}	Dimensional variation of rolling moment with differential flap
L_{DDF}'	B-matrix L_{DDF} derivative - $L_{DDF}' = (L_{DDF} + A_1 N_{DDF})/(1 - A_1 B_1)$
m	Pendulum mass
M	Carriage mass in pendulum example
M_u	Dimensional variation of pitching moment with speed
M_u'	A-matrix M_u derivative - $M_u' = M_u + M_\alpha Z_u'$
M_α	Dimensional variation of pitching moment with angle of attack

Symbol	Definition
$M_{\dot{\alpha}}$	Dimensional variation of pitching moment with rate of change of angle of attack
M_{α}'	A-matrix M_{α} derivative - $M_{\alpha}' = M_{\alpha} + M_{\dot{\alpha}} Z_{\alpha}'$
M_q	Dimensional variation of pitching moment with pitch rate
M_q'	A-matrix M_q derivative - $M_q' = M_q + M_{\dot{\alpha}} Z_q'$
M_{DC}	Dimensional variation of pitching moment with canard angle
M_{DC}'	B-matrix M_{DC} derivative - $M_{DC}' = M_{DC} + M_{\dot{\alpha}} Z_{DC}$
M_{DF}, M_{DF}'	Same as M_{DC}, except with respect to flap angle
M_{DST}, M_{DST}'	Same as M_{DC}, except with respect to strake angle
N_P	Dimensional variation of yawing moment with roll rate
N_p'	A-matrix N_p derivative - $N_p' = (N_p + B_1 L_p)/(1 - A_1 B_1)$
N_r	Dimensional variation of yawing moment with yaw rate
N_r'	A-matrix N_r derivative - $N_r' = (N_r + B_1 L_r)/(1 - A_1 B_1)$
N_β	Dimensional variation of yawing moment with sideslip
N_β'	A-matrix N_β derivative - $N_\beta' = (N_\beta + B_1 L_\beta)/(1 - A_1 B_1)$
N_{DR}	Dimensional variaton of yawing moment with rudder
N_{DR}'	B-matrix N_{DR} derivative - $N_{DR}' = (N_{DR} + B_1 L_{DR})/(1 - A_1 B_1)$
N_{DDF}	Dimensional variation of yawing moment with differential flap

Symbol	Definition
N_{DDF}'	B-matrix N_{DDF} derivative - $N_{DDF}' = (N_{DDF}+B_1 L_{DDF})/(1-A_1 B_1)$
p	Parameter, also roll rate
q	Pitch rate
r	Yaw rate
r	Command input to closed-loop control system
R	Rudder
s	Laplace variable
S	Diagonal Matrix of singular values
ST	Strake
SF	Symmetric flap
t	Time
T	Closed-loop system transfer function matrix (s-domain); also sample time for sampled-data systems
u	Control system input, also left singular vector
u_a	Control input to control system (augmentation)
u_c	Control input to continuous part of a sampled-data system
u_d	Control input to digital part of a sampled-data system
u_s	Scaled control system input
U	Matrix of left singular vectors in singular value decomposition
U_1	Airplane steady-state speed
V	Matrix of right singular vectors in singular value decomposition

Symbol	Definition
W	Disturbance vector (s-domain)
W,X,Y,Z	General matrices used in examples
x	State vector
x_a	State vector of control system (augmentation)
x_c	State vector of continuous part of a sampled-data system
x_d	State vector of digital part of a sampled-data system
x_p	State vector of the plant
X_u	Dimensional variation of longitudinal force with speed
X_α	Dimensional variation of longitudinal force with angle of attack
X_q	Dimensional variation of longitudinal force with pitch rate
X_{DC}	Dimensional variation of longitudinal force with canard angle
X_{DF}	Dimensional variation of longitudinal force with flap angle
X_{DST}	Dimensional variation of longitudinal force with strake angle
y	Output vector
y_a	Output vector of control system (augmentation)
y_c	Output vector of continuous part of a sampled-data system

Symbol	Definition
y_d	Output vector of digital part of a sampled-data system
y_p	Output vector of the plant
Y_p	Dimensional variation of side force with roll rate
Y_p'	A-matrix Y_p derivative - $Y_p' = Y_p/U_1$
Y_r	Dimensional variation of side force with yaw rate
Y_r'	A-matrix Y_r derivative - $Y_r' = Y_r/U_1 - 1$
Y_β	Dimensional variation of side force with sideslip
Y_β'	A-matrix Y_β derivative - $Y_\beta' = Y_\beta/U_1$
Y_{DR}	Dimensional variaton of side force with rudder
Y_{DR}'	B-matrix Y_{DR} derivative - $Y_{DR}' = Y_{DR}/U_1$
Y_{DDF}	Dimensional variation of side force with differential flap
Y_{DDF}'	B-matrix Y_{DDF} derivative - $Y_{DDF}' = Y_{DDF}/U_1$
z	Digital domain variable: $z=e^{-sT}$
Z	Measurement vector (s-domain)
Z_U	Dimensional variation of lift with speed
Z_u'	A-matrix Z_u derivative - $Z_u' = Z_u/(1-Z_{\dot{\alpha}})$
Z_α	Dimensional variation of lift with angle of attack
$Z_{\dot{\alpha}}$	Dimensional variation of lift with rate of change of angle of attack
Z_α'	A-matrix Z_α derivative - $Z_\alpha' = Z_\alpha/(1-Z_{\dot{\alpha}})$
Z_q	Dimensional variation of pitching moment with pitch rate
Z_q'	A-matrix Z_q derivative - $Z_q' = (U1+Z_q)/(U1-Z_{\dot{\alpha}})$

Symbol	Definition
Z_{DC}	Dimensional variation of pitching moment with canard angle
Z_{DC}'	B-matrix Z_{DC} derivative - $Z_{DC}' = Z_{DC}/(1-Z_{\dot{\alpha}})$
Z_{DF}, Z_{DF}'	Same as Z_{DC}, except with respect to flap angle
Z_{DST}, Z_{DST}'	Same as Z_{DC}, except with respect to strake angle
α	Angle of attack; also dummy variable in pendulum example
β	Sideslip angle
δ	Integral of θ in pendulum example
δ_{ij}	Kroeneker delta of dimension i x j
ζ	Augmented-system control input for pendulum example
θ	Pitch attitude angle, also pendulum angle
λ	Eigenvalue
μ	Force on carriage in pendulum example
ξ_1, ξ_2	States in pendulum example
ρ_p, ρ_z	Compensator pole- and zero- locations in pendulum example
σ_i	i-th Singular value
$\underline{\sigma}, \bar{\sigma}$	Minimum and maximum singular value respectively
$\underline{\sigma}_{min}$	Minimum of the plot of $\underline{\sigma}$ with respect to ω ($\underline{\sigma}$-plot)
τ	Dummy integration variable
ϕ	State transition matrix for digital systems; also phase perturbation; also bank angle
ψ	Servo state in pendulum example
ω	Frequency, in rad/sec

LIST OF SYMBOLS AND ACRONYMS, continued

Symbol	Definition
Γ	Discrete control power matrix
Δ	"Small change in"
Θ	The zero vector (origin in n-space)
Σ	Indicates summation; also, non-singular submatrix of S in the singular value decomposition
Φ	Laplacian of the state transition matrix, $(Is-A)^{-1}$
Ψ	Integral of the state transition matrix
Ω	Z-transform of the discrete state transition, $(Iz-\Phi)^{-1}$

Operators:

Σ	Summation
$\|A\|$	Matrix or vector Euclidean norm of A
$\angle A$	Argument or angle of A
$\rho(A)$	Eigenvalue spectrum of A
$\bar{\sigma}(A), \underline{\sigma}(A)$	Maximum, minimum singular value of A (equivalent to $\|A\|$)
diag(x)	Diagonal matrix with elements x
$(A)_i, (A)_{ij}$	i,j partition of A
tr(A)	Trace of A (sum of the diagonal elements)
$\partial a/\partial x$	Partial derivative of a with respect to x
$\partial a/\partial X$	Matrix whose elements are the partial derivatives of a with respect to the corresponding element of X
Re(A)	Real part of a complex matrix A
Imag(A)	Imaginary part of a complex matrix A

Symbol	Definition
A^*	Conjugate-transpose of a matrix A (equivalent to transpose if A is real)

A note on notation:

The rules for notation in this document are as follows: scalars are generally represented as lower case letters. Lower case boldface characters represent vectors; upper case boldface characters are the Laplace transfer functions of time domain vectors. Upper case letters generally represent matrices. Unless context permits omission, a matrix transfer function A is denoted by an A(s).

Acronym	Definition
AFCS	Automatic flight control system
GM	Gain margin
KU	University of Kansas
MIMO	Multi-input, multi-output
NASA	National Aeronautics and Space Administration
PM	Phase margin
SISO	Single-input, single-output
SVA	Singular value analysis
FRL	Flight Research Lab

1. INTRODUCTION

Most designers of automatic flight control systems (AFCS) model the aircraft being controlled by assuming that the aircraft's motion can be described by a set of linear, time-invariant, differential equations. These equations can be written in matrix form as

$$\dot{x} = Ax + Bu \,, \tag{1.1}$$

where x is a time-varying vector of aircraft states, such as roll rate and bank angle, and u is a time-varying vector of commanded control positions, such as aileron command and rudder command. A and B are the 'dynamic matrices,' the constant coefficients of the differential equations.

The commanded control positions u can come either from a pilot or from an AFCS. In the latter case the commands can often be modeled as feedbacks of the aircraft states:

$$u = -Cx. \tag{1.2}$$

This system is depicted schematically in Figure 1.1. Equations (1.1) and (1.2) combine to form the closed-loop dynamic system, whose performance and stability characteristics can be very different from those of the airplane described by Equation (1.1) alone, which is called the open-loop system.

In general, the coefficients of the dynamic matrices (A and B) are not exactly known. Errors in these coefficients are called modeling errors and are due to two sources: the linearization of the nonlinear aircraft dynamics (which yields Equation 1.1), and

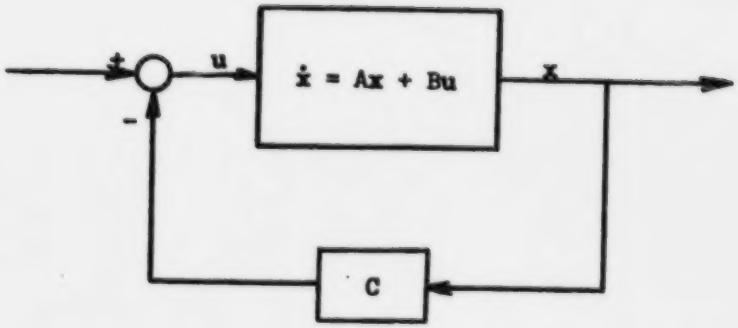


Figure 1.1: Block diagram of a basic feedback control system

uncertainty about the actual dynamic characteristics of the airplane. Control system design is intimately concerned with the effect of modeling errors on the performance and stability of the closed-loop dynamic system. Methods have been developed to deal with modeling errors during both design and verification of a flight control system.

During the design phase, Bode methods [2] have traditionally been used to insure system "robustness" (in the form of gain margin and phase margin)--that is, to insure that even if the actual aircraft is different than the one for which the control system was designed, the closed loop system will still be stable and have good performance. Concurrently, efforts are made to identify the model parameters as accurately as possible, so that the system can be designed based on accurate information.

During the verification of an AFCS, two methods are usually used to test the system in the face of modeling errors. Firstly, the system is analyzed assuming some set of parameter variations, to see if its stability and robustness are adversely affected. Secondly, and usually subsequently, the actual system is flight tested in a careful and systematic way to assess its performance.

The methods described above have worked extremely well in the past, and are still applied extensively in present control system design. However, modern aircraft are becoming more complex. For instance, the highly augmented X-29 research vehicle is a 48th order system longitudinally. Another example is the proposed F-8 oblique

wing airplane, which will be a fully coupled 6-degree of freedom system, and will require a very high degree of augmentation. Another complication for classical design resides in the fact that modern control methods often result in many feedback paths. These problems, complexity and multiple feedback paths, are not addressed by the classical techniques for insuring stability and robustness. This fact is apparent, once again, in both the design and verification phases of AFCS design.

In the design phase, for instance, Bode techniques are only strictly applicable to single-input single-output (SISO) systems [1]. Designers of multiloop systems cannot use Bode measures without some misgivings about their validity. Also, identifying the dynamic model accurately is more difficult for complex aircraft, and requires costly flight testing. For open-loop unstable vehicles such as the X-29, such flight testing cannot occur unless an active control system is augmenting the airplane!

In the verification phase, analyzing the control system in the face of selected parameter variations is very time consuming, and invariably incomplete, when the system is of very high order. This is because the number of parameters and combinations of parameters which can be varied is very high. Finally, flight testing for verification can be unsafe, because the control systems being verified may be necessary to insure adequate stability and controllability. Early airplanes often still flew if the control system was flawed; this is no longer necessarily true. The result

is the necessity for costly, lengthy verification before any flight testing can occur.

Thus many of the accepted techniques for insuring the stability and robustness of AFCS have drawbacks in the context of modern control design. Fortunately, methods have been developed to deal with these problems. The goal of these efforts is often to extend the well-understood classical concepts so that they can also be applied to multi-input multi-output (MIMO) controllers. Many of these methods are based on the singular values of the control system return difference matrix, and on the gradients of these singular values. Robustness measures, some of which parallel the Bode measures of phase and gain margin very closely, have been developed in [3] through [6]. Methods for designing robust controllers using singular values are presented in [4], [7], and [8]. In [8], Mukhopadyay and Newsom present a design method which uses singular values and singular-value gradients.

In [1], Herrera et al. use the results of [8] to develop a technique to extend classical sensitivity methods to modern MIMO control systems. By using singular values and their gradients, a control system, once designed, can be analyzed to determine those parameters, out of the many that describe the control system, which affect the stability and robustness of the closed loop system most dramatically. Identifying these parameters eliminates the need to identify all the system parameters very accurately. This relieves somewhat the necessity for intense parameter identification efforts.

It also allows parameter-variation type testing to be done in a systematic and complete way.

The work of Herrera et al. was performed under NASA contract NCC2-293, which was awarded to the University of Kansas for the development of a sensitivity analysis technique for continuous multiloop systems. The technique resulting from phase one of this contract is documented in [1] and [9] and consists of computing the singular values of the return difference matrix and the gradients of these singular values with respect to model parameters. During phase one, this technique was applied to several low-order systems, to determine the characteristics and viability of the method.

This report details the work done during phase two of NCC 2-293. The primary goals of this phase were to install the software for the singular value analysis (SVA) technique at Ames Research Center, Dryden Flight Research Facility, to test it on real systems, and to familiarize Dryden personnel with the technique and with singular values in general. To fulfill these goals, several improvements had to be made to the SVA developed at KU. First of all, the analysis had to be extended so that digital systems could be analyzed. In addition, options were added to the software to deal with problems which arise when one is analyzing complex systems. Finally, work was done to make the software easy to use.

This report is organized as follows: Section 2 gives a theoretical background and mathematical derivation of the SVA.

Section 3 extends the derivation of the singular-value gradients to include digital systems. The SVA technique is illustrated in Section 4 for a simple example of an inverted pendulum. Section 5 presents results obtained for a real system (the X-29) at NASA-Dryden. Appendix A contains those derivations which are considered too involved for the text; Appendix B tabulates numerical data for the X-29; and Appendix C describes the SVA program.

2. BACKGROUND

This section reviews the results of Herrera ([1] and [9]). Section 2.1 reviews the Bode technique, and details the problems which motivate the use of singular values; these problems were discussed briefly in the introduction. Section 2.2 then gives a mathematical description of what singular values represent in the context of control system design. Section 2.3 elaborates on the use of singular values as measures of robustness, discussing their shortcomings and giving some alternate ways to apply them. Finally, Section 2.4 describes singular value gradients, their derivation and application. Qualitative results from [1] will give helpful insight into the SVA in this and future sections.

2.1 THE NEED FOR A MULTILOOP ANALYSIS METHOD

Figure 2.1 is a block diagram of a typical single-input, single-output (SISO) control system. The Laplace domain transfer function g(s) represents the dynamics of the aircraft being controlled. h(s) is the control system transfer function; it models all compensation and feedback gains. The transfer function l(s) represents any disturbances or variations in the system, either due to plant parameter variations or controller parameter variations.

The closed loop transfer function of this system is

$$\frac{u}{r}(s) = \frac{1}{1 + lgh(s)} \qquad (2.1)$$

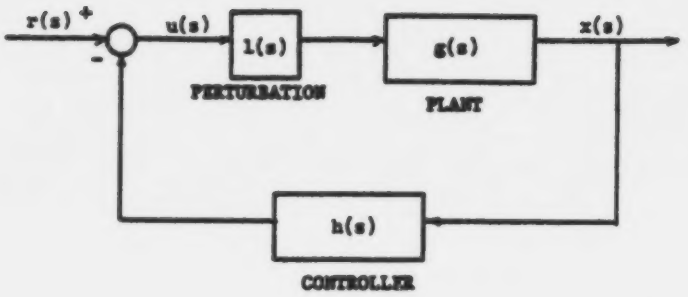


Figure 2.1: Block diagram of a single-input single-output control system with a gain and phase perturbation at the input

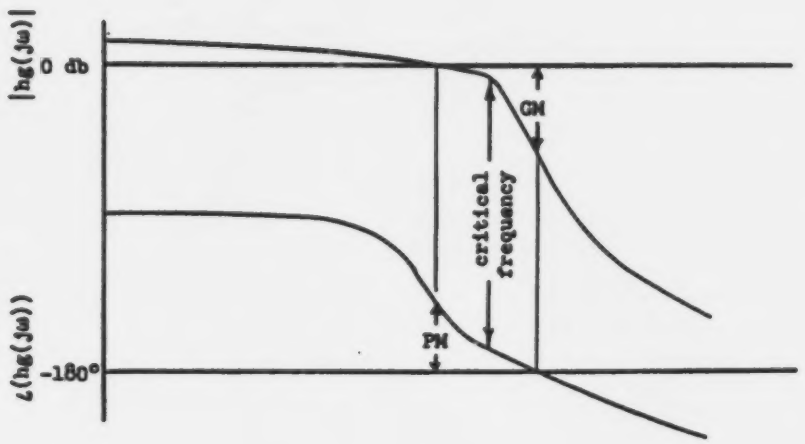


Figure 2.2: Example of a Bode plot with misleading gain and phase margins

where u(s) and r(s) are the Laplace transforms of the input to the open loop system and the input to the closed loop system, respectively. Both u(s) and r(s) are scalars. Obviously, u(s) will be unbounded for a bounded input if

$$lhg(s) = -1. \tag{2.2}$$

The Bode method checks for closeness to instability by applying a sinusoidal input, $r(t)=\sin(\omega t)$. [2] shows that this yields the criterion

$$|lhg(j\omega)| = 1, \quad \angle lhg(j\omega) = -180 \text{ degrees}, \tag{2.3}$$

where $j = \sqrt{-1}$.

When this condition is fulfilled, the system has a pure resonance, and thus is on the boundary between the stable and unstable regions. Closeness to instability is judged by giving 1 the special form $1=ke^{j\phi}$, which represents a disturbance in both the gain and phase of the system:

$$\sin(\omega t) = \text{Imag}[e^{j\omega t}] ,$$

$$k\sin(\omega t + \phi) = \text{Imag}[e^{j\omega t} \cdot ke^{j\phi}] . \tag{2.4}$$

Gain margin (GM) and phase margin (PM) are then defined as follows:

GM = the value of k which causes condition (2.3) to be met if $\phi=0$ (no phase change).

PM = the value of ϕ which causes condition (2.3) to be met if $k=1$ (no gain change).

The definitions for GM and PM will only be fulfilled at certain frequencies. The GM criterion can be satisfied only at frequencies where $\angle lhg(j\omega) = -180$, and the PM criterion can only be satisfied at

frequencies where $|lhg(j\omega)| = 1$ (or 0 dB). To check for these points, one must plot $lhg(j\omega)$ for varying ω. The most popular way to do this is using a Bode plot, but Nyquist diagrams and Nichols charts are also used for various applications. Figure 2.2 illustrates the definitions of GM and PM on a Bode plot.

One can apply the technique above to multiloop systems, but only in a limited way. Figure 2.3 illustrates how this is done. Consider the two-loop control system shown in 2.3(a), where G(s) now represents a two-by-two matrix of transfer functions, all of whose elements may be nonzero. This system can be analyzed using traditional Bode methods by introducing a perturbation $l=ke^{j\phi}$ into one loop of the system. The analysis then procedes as above, with the assumption that the other loop is a fixed part of the 'open-loop' dynamics of the plant, as shown in 2.3(b).

Two possibilities are ignored by this type of analysis. The first is the possibility that the perturbation l may actually destabilize the h_1 loop in Figure 2.3. The second is that if the h_1 loop varies in some way (gain or phase), and is not fixed as assumed, the outer loop gain and phase margins may change drastically.

Taking these kind of possibilities into account is analagous to avoiding a steep slope in either the gain or phase curve of a single-loop Bode plot. If, for instance, the phase curve of a SISO system is steep at the frequency of a shallow gain crossover, a slight change in gain could cause the phase margin to change

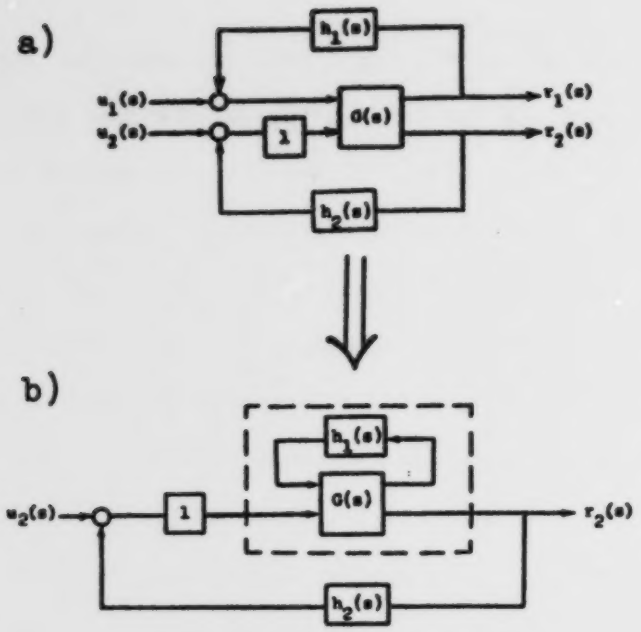


Figure 2.3: Diagram of classical technique for studying perturbations in multi-input multi-output systems

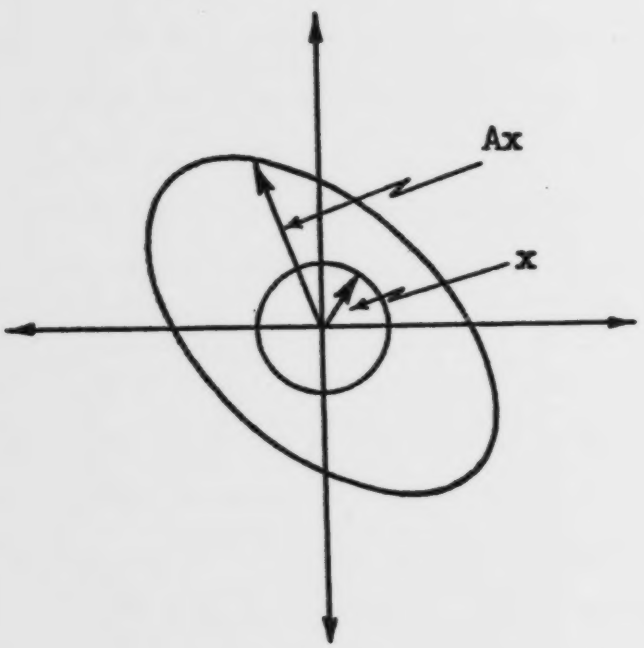


Figure 2.4: 2-dimensional illustration of a matrix as a vector transformation

drastically. This situation is illustrated in Figure 2.2 (this Bode plot may not actually be physically realizable; it is used here only as a graphic illustration). Thus a system is not necessarily robust even if it has acceptable gain and phase margins.

Apparently, a good measure of nearness to instability would take into account <u>simultaneous</u> changes in gain and phase in <u>all</u> the loops. In the next section, such a measure will be presented, after some necessary mathematical concepts are introduced.

2.2 VECTOR AND MATRIX NORMS, SINGULAR VALUES, AND THEIR APPLICATION

The concept of stability in the scalar case presented above depended on the idea that the input u(s) to the plant should remain bounded in the closed loop system. For an n-dimensional vector of inputs, this boundedness criterion can be extended by using the vector Euclidian norm, defined as

$$\begin{aligned} \|x\| &= \sqrt{\sum_{i=1}^{n} x_i^2} \\ &= \sqrt{x^T x} \, . \end{aligned} \tag{2.5}$$

$\|x\|$ can be interpreted as the length of the vector x in n-space. Complex vectors utilize a slightly different definition to yield a "length". If $x = u + vj$, then the norm is defined as

$$\begin{aligned} \|x\| &= \sqrt{\sum_{i=1}^{n} (u_i + v_i j)(u_i - v_i j)} \\ &= \sqrt{x^* x} \, , \end{aligned} \tag{2.6}$$

where x^* is the conjugate-transpose of x.

The analysis to be presented also requires the "size" of a matrix to be quantified in some way. This is easily done by thinking of a matrix A as a linear transformation, which transforms any compatibly dimensioned vector x into a vector Ax which has been stretched and rotated in n-space. The "size" of A, then, can be thought of as the maximum or minimum possible change in size, or "stretching factor," that A can cause as a transformation. This is in fact how the matrix Euclidian norm is defined; mathematically this is written

$$\|A\| = \begin{matrix}\max \\ \text{or} \\ \min\end{matrix} \left(\frac{\|Ax\|}{\|x\|}\right) \text{ for all } x \text{ ,}$$

$$\text{or} \qquad \|A\| = \begin{matrix}\max \\ \text{or} \\ \min\end{matrix} \left(\|Ax\|\right) \text{ for all } x \text{ such that } \|x\| = 1 \text{ .} \tag{2.7}$$

Figure 2.4 shows the interpretation of this matrix norm in 2-space. The vector x is allowed to vary in any way, as long as its length remains unity. As it traces a circle, Ax traces an ellipse. The maximum and minimum lengths of the resulting vector are the Euclidean norms of the matrix A.

When the experiment described above is performed on an n-dimensional matrix a hyperellipsoid (or a degenerate thereof, depending on the size and rank of the matrix) always results. Now suppose that in Figure 2.4 we are able to break the transformation A into a rotation of the axes and a standard equation for an ellipse,

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1. \qquad (2.8)$$

It is apparent that a and b, which are half the major and minor axes of the ellipse, are the norms of A that we defined in Equation (2.7). The singular value decomposition breaks the transformation A up in exactly this way. The basic theorem of the singular value decomposition is that any matrix A can be represented as USV^*, where U and V are unitary matrices (which means that their transformations yield no change in length, only rotations), and S is a diagonal matrix of singular values, which are denoted by σ_i, i=1..n. The singular values are half the lengths of the axes of the hyperellipsoid created by $\{Ax, x: \|x\| = 1\}$, and the maximum and minimum singular values, $\bar{\sigma}$ and $\underline{\sigma}$, respectively, are the matrix Euclidian norms. Proofs of both the singular value decomposition theorem and the fact that singular values correspond to norms are presented in Appendix A, taken from References [10] and [11].

We now have all the necessary tools to conduct a multiloop frequency response. Figure 2.5 represents a MIMO system, in which U(s), R(s), and X(s) are vectors and L(s), G(s), and H(s) are matrices. The equation for the matrix of transfer functions for the closed loop system has a form analogous to Equation (2.1),

$$\begin{aligned} U(s) &= [I + HGL(s)]^{-1} R(s) \\ &= TR(s). \end{aligned} \qquad (2.9)$$

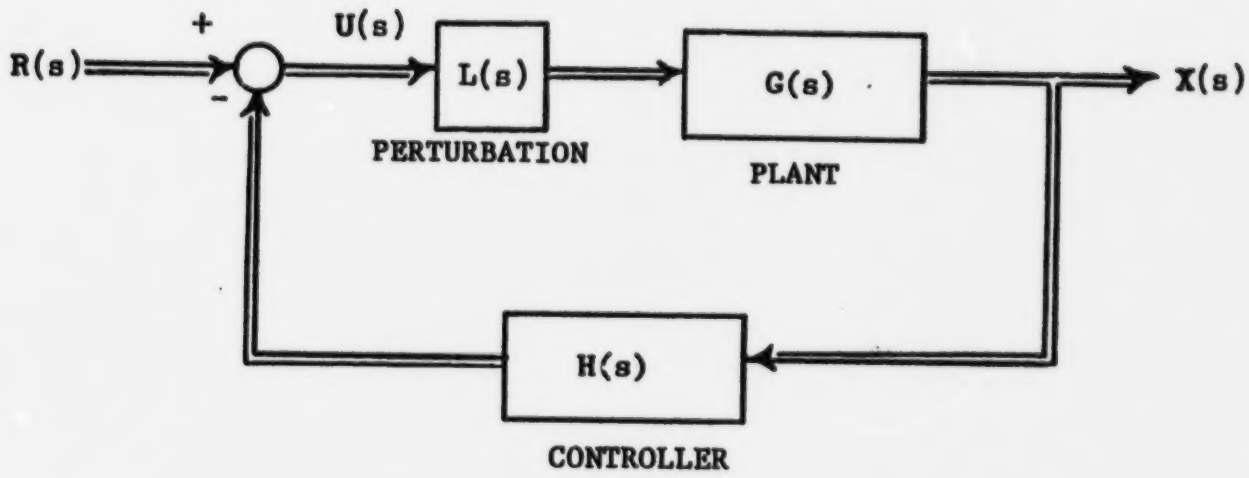


Figure 2.5: Block diagram of a multi-input multi-output control system with a perturbation matrix at the input

In this case, however, boundedness will be defined with the help of norms. The "size" of the input-output relation is given by

$$\frac{\|U(s)\|}{\|R(s)\|} = \frac{\|TR(s)\|}{\|R(s)\|} . \tag{2.10}$$

The maximum of (2.10) translates directly into singular values using the definition in equation (2.7):

$$\max \frac{\|TR(s)\|}{\|R(s)\|} = \|T\| = \bar{\sigma}(T) = \bar{\sigma}[(I + HGL(s)^{-1}] = \frac{1}{\underline{\sigma}[I + HGL(s)]} , \tag{2.11}$$

where a special property of singular values has been used for the last equality. This property is proven in Appendix A. Now we can say that $\|U\|$ will be unbounded when the minimum singular value of (I+HGL), which is called the return difference matrix, is zero. This condition can be checked across the frequency range [3] by testing the criterion

$$\underline{\sigma}[I + HGL(j\omega)] = 0 \tag{2.12}$$

for all ω of interest. When this equation is satisfied, a pure resonance at ω exists. This resonance can be interpreted as a pole of the closed loop system on the imaginary axis, which means the system is on the "stability boundary". It is important to note that singular values, because they are the absolute "lengths" of a transformation, will always be nonnegative. So if the original, unperturbed closed-loop system is not on the stability <u>boundary</u> represented by Equation (2.12), the minimum singular value of the return difference matrix will be positive, whether the system is stable or unstable. This situation is analogous to that presented by the Bode plot, which also gives no indication of stability.

If the unperturbed system is stable, norms can be used to represent nearness to instability. To this end we take L to be a diagonal matrix whose elements are analagous to the single loop perturbation l.

$$L = \text{diag}[k_1 e^{j\phi_1}, k_2 e^{j\phi_2}, \ldots k_n e^{j\phi_n}]. \qquad (2.13)$$

k_i and ϕ_i (i=1,n) may vary independently in any way. The system will remain stable if the following criterion, based on (2.12), is met: L must be smaller than the smallest matrix J for which

$$\underline{\sigma}[I + HGJ(j\omega)] = 0 \qquad (2.14)$$

at some ω. This is analogous to the gain and phase margin concepts in SISO systems, except that in the case of singular values, all the gains and phases may vary simultaneously to achieve the stability boundary. It can be shown that criterion (2.14) will be met if the following equation is true:

$$\underline{\sigma}[I + HG(j\omega)] > \bar{\sigma}(L^{-1} - I) \; . \qquad (2.15)$$

See Appendix A for the proof of this fact. [3] also shows that it is possible to rewrite the right-hand side of the above equation in terms of the maximum values of k_i and ϕ_i in the matrix L:

$$\bar{\sigma}(L^{-1} - I) = \sqrt{(1 - \frac{1}{k_{max}})^2 + \frac{2}{k_{max}}(1 - \cos\phi_{max})} \; . \qquad (2.16)$$

To judge nearness to instability, combine Equation (2.15) and (2.16) through the following steps:

1. Compute $\underline{\sigma}[I+HG(j\omega)]$ for various ω's. The resulting plot is called a '$\underline{\sigma}$-plot.' It traces the nearness to singularity of the return difference matrix, and thus the system's robustness to

perturbations, with changing frequency. Note that the perturbation L need not be known to compute the $\underline{\sigma}$-plot.

2. Determine the minimum of the $\underline{\sigma}$-plot. This will be denoted by $\underline{\sigma}_{min}$. $\underline{\sigma}_{min}$ occurs at the frequency at which the system is nearest to the stability boundary.

3. If $\bar{\sigma}[(I + L)^{-1}] < \underline{\sigma}_{min}$, the system will remain stable. Thus if all k_i and ϕ_i are properly bounded by Equation (2.16), the system will remain stable.

Equation (2.16) can also be presented as a "universal diagram for gain and phase margin evaluation" [3]; this diagram is shown in Figure 2.6. An example will illustrate its use. If the smallest $\underline{\sigma}(I+HG)$ for a system is .6, then the closed-loop system will tolerate simultaneous gain and phase changes of -1.5 dB to 5.3 dB, and -30 deg to +30 deg, respectively, in all input loops. In a classical sense, when either gain or phase is changed while the other is held constant, the margins are -4.2 dB and +8 dB or ±35 deg, respectively. The latter results can also be obtained by alternately setting gain and phase to zero in Equation (2.16), yielding the following Equations [5,6]:

$$GM = \frac{1}{1 \pm \underline{\sigma}_{min}}, \tag{2.17a}$$

$$\left.\begin{aligned} PM &= \pm \cos^{-1}\left(1 - \frac{(\underline{\sigma}_{min})^2}{2}\right) \\ &= \pm 2\sin^{-1}\left(\frac{\underline{\sigma}_{min}}{2}\right) \end{aligned}\right\} \quad \text{if } \underline{\sigma}_{min} < 2,$$

$$\text{and } PM = \pm 180 \text{ if } \underline{\sigma} > 2 \tag{2.17b}$$

It is important to note that Equation (2.15) is a conservative condition and that it is possible to construct a matrix L which

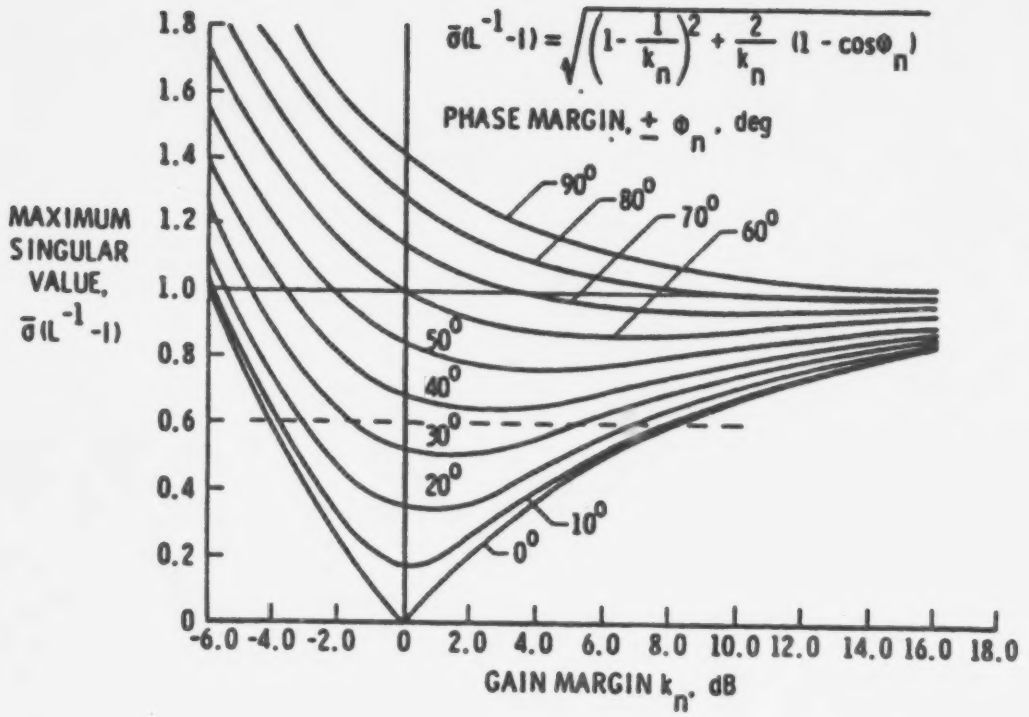


Figure 2.6: Universal diagram for multiloop gain-phase margin evaluation (Reference [3])

violates it, yet fails to drive the system all the way to the stability boundary. In other words, the minimum of the $\underline{\sigma}$-plot is the size of the smallest L which will drive the system to the boundary [19], but there are many matrices of the same size that do not drive it that far. However, no matrix of disturbances which is "smaller" than the matrix in Equation (2.15) will destabilize a stable system. So (2.16) is a guaranteed allowable limit for all the gain and phase changes in the matrix L.

It is instructive at this point to give a SISO example to illustrate the similarities and differences between classical and singular-value-based techniques. For a SISO system the return difference matrix is the scalar $[1 + hg(j\omega)]$. The singular value of this scalar is simply its magnitude. The frequency response of the system, on the other hand, is $hg(j\omega)$, which is a complex number for each ω. Therefore, if the frequency response is plotted on a Nyquist diagram, as in Figure 2.7, the singular value is simply the distance from the instability point, -1+0j, to the Nyquist plot! This characteristic can be seen in Figure 2.9, which is the $\underline{\sigma}$-plot corresponding to Figure 2.7.

GM on a Nyquist plot is taken at the point(s) where the curve crosses the real axis to the left of the origin (i.e., $\angle\, hg(j\omega) = -180$ degrees). PM is taken at the point(s) where the curve crosses the unit circle (i.e. $|hg(j\omega)| = 1$). In Figure 2.7, the GM occurs at $\omega = 2.6$ rad/sec and is .34 or -9.3 dB. A PM of +38 degrees occurs at $\omega = 19$ rad/sec.

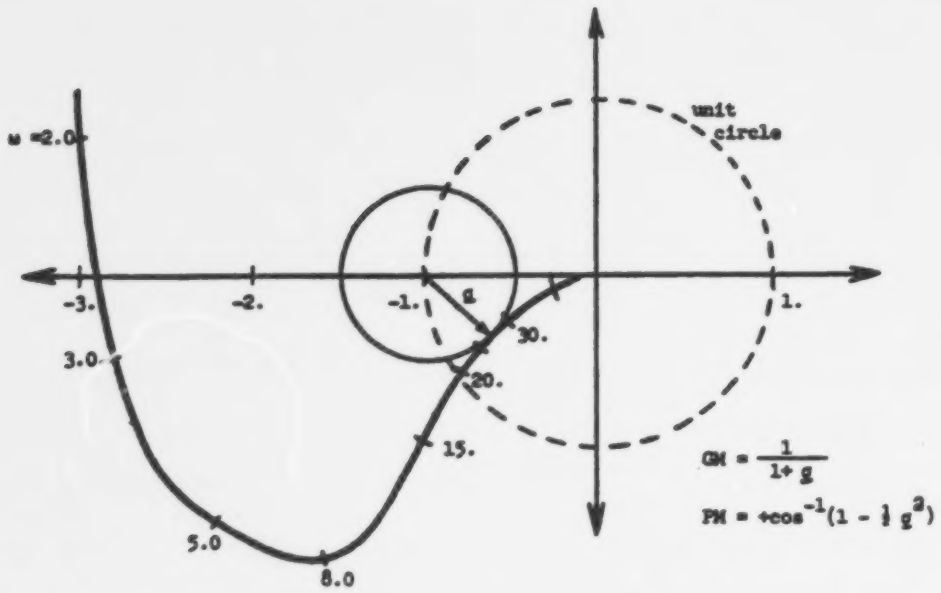


Figure 2.7: Illustration of the minimum singular value of the return difference matrix on a Nyquist plot (SISO)

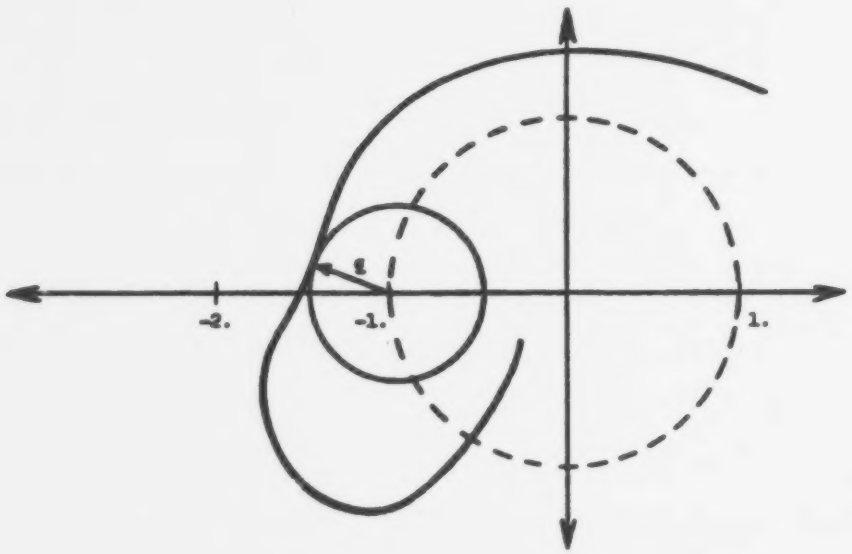


Figure 2.8: Example of a Nyquist plot with the same minimum singular value

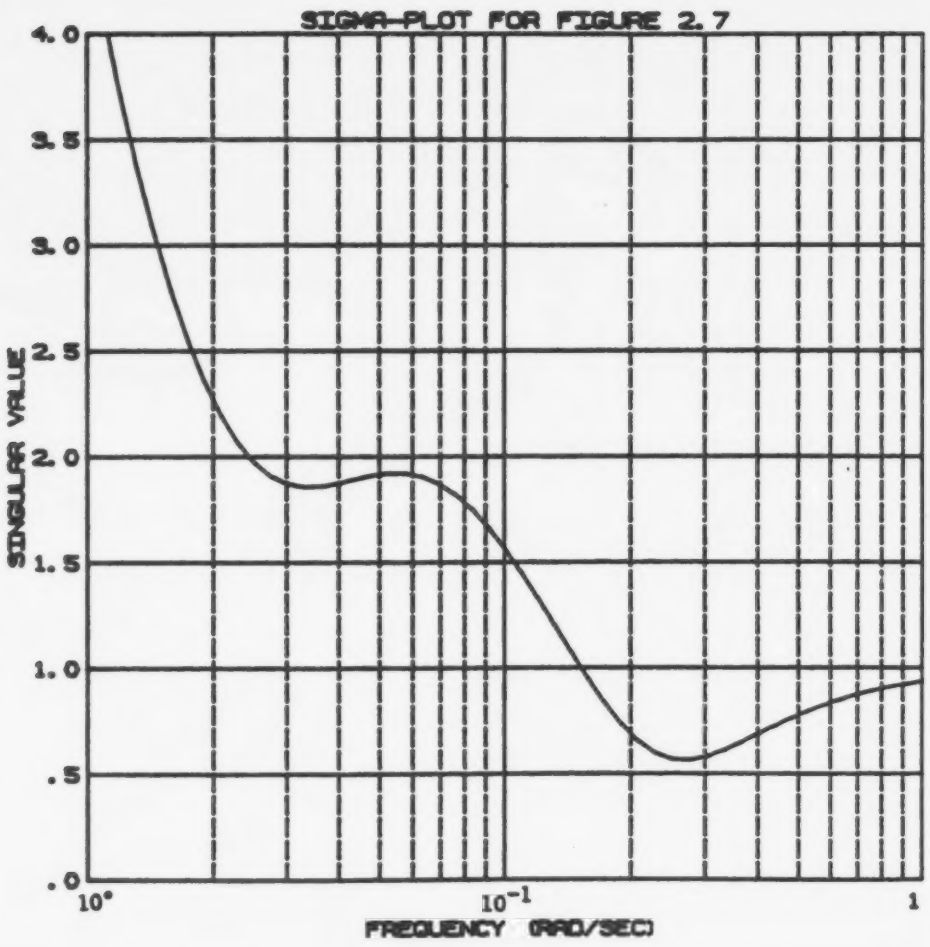


Figure 2.9: Singular-value plot for the system whose Nyquist plot appears in Figure (2.7).

In contrast, the singular value analysis calculates simultaneous GM-PM limits by finding the point on the curve which is nearest to the stability boundary of -1+0j. In Figure 2.7, this occurs at 27.5 rad/sec. This point can also be found in Figure 2.9; it is simply the absolute minimum of the curve. It can be seen in Figure 2.7 that the system will be driven to -1+0j at 27 rad/sec by a gain variation of 1.5 or +3.7 dB combined with a phase variation of 32 degrees. The "size" of this variation corresponds exactly to the minimum distance from the curve to the -1 point, and can be found, using Equation (2.16), Figure 2.7, or Figure 2.9, to be .56.

Equations (2.17a) and (2.17b) arise by drawing a circle of length $\underline{\sigma}_{min}$ around the -1 point. This must be done because the SVA provides no information about the direction one must go to get to the stability boundary. For instance, the curve in Figure 2.8 would yield the same $\underline{\sigma}_{min}$ as Figure 2.7. So to guarantee that the system will remain stable in the face of any simultaneous variations (NOT just the one mentioned above) the gain and phase margins must be valid for any point on the circle of radius $\underline{\sigma}_{min}$. Thus the "classical" GM and PM occur where this circle intersects the real axis and the unit circle, respectively. These values will not necessarily match those obtained using the true classical definitions of GM and PM. As can easily be seen in Figure 2.7, they will be conservative. The conservativeness is much less easy to interpret in the MIMO case, because Figure 2.7 cannot be

constructed. Only the information one can glean from figures like Figure 2.9 is available.

In an effort to reduce the conservativeness of the gain and phase margins predicted by Equations (2.16) and (2.17), References [3] and [6] studied the use of eigenvalues instead of singular values. The minimum eigenvalue of the return difference matrix will also be zero when the system is on the stability boundary. Thus a plot of $\underline{\lambda}$ traces the matrix's nearness to singularity in complex space. However, no rule like Equation (2.16) is available for eigenvalues unless the disturbance matrix L is taken to contain only 'uniform' uncertainties. For the uncertainties to be uniform, the matrix L must have the form $ke^{j\phi}[I]$; that is, the gains and phases along the diagonal of L must vary together, instead of independently. Reference [6] shows that the universal gain and phase plot in Figure 2.6 can be applied using the minimum eigenvalue under the restriction that the gain and phase variations be uniform as described here.

Reference [1] is an extensive study of $\underline{\sigma}$-plots, and includes several examples which compare MIMO ($\underline{\sigma}$-plot) results to SISO (Bode plot) results. Some of the characteristics of $\underline{\sigma}$-plots discovered in [1], which will become evident in future chapters of this report, are

1) The effects of poles or "modes" of the system can be seen in $\underline{\sigma}$-plots. Usually a dip or peak occurs near the frequency of each pole of the closed-loop system.

2) As discussed above, it is not possible to determine if a system is stable or unstable by looking only at $\underline{\sigma}$-plots. Only the

nearness to the stability boundary, and a conservative estimate of the minimum gain and phase changes necessary to drive the system there, can be determined.

3) A plot of the eigenvalues of the return difference matrix on the same graph as the $\underline{\sigma}$-plot is a useful addition to the analysis. Because it is based on a relaxed criterion (that all gains and phases must vary together), it will always lie above the $\underline{\sigma}$-plot. The conservativeness of the $\underline{\sigma}$-plot can be qualitatively judged by the distance between the $\underline{\sigma}$-plot and the eigenvalue-plot.

4) Some rules of thumb for singular values are:

> For a system which will have a multiloop gain margin of +10dB and a multiloop phase margin of ±40 degrees, keep the $\underline{\sigma}$-plot above .684.
>
> A $\underline{\sigma}$ of 1 represents a system which is robust in the optimal sense; it has a gain margin of $+\infty$ and a phase margin of ±60 degrees.

Because of the way multiloop gain and phase margins are defined, the use of a +10dB and ±40 degrees rule of thumb may be too stringent. This rule was probably developed in part to absorb some of the under-conservatism of Bode plots, which do not take simultaneous changes into account even for single-loop systems. It has been found that a minimum $\underline{\sigma}$ of .4 - .5 usually still represents a good design.

2.3 ELABORATION OF THE CHARACTERISTICS OF SINGULAR VALUES

The analysis presented in Section 2.2 has many attractive properties, but recent work indicates that it contains many flaws and complications. These will be discussed in detail in this section so that anyone using the analysis presented in this paper will be fully aware of the shortcomings and subtleties of singular value analysis.

Several of the complications involved with singular values stem from the unstructured nature of the matrix L in Equations (2.14) and (2.15). Although a special form is chosen for L in Equation (2.13), the derivation of Equation (2.15) is valid for any matrix L (see Appendix A). What this means is that even if L has non-zero off-diagonal elements, one can still gaurantee stability as long as condition (2.15) is met. Non-zero off-diagonal elements in L constitute cross-feed perturbations between feedback loops. These are not part of the definitions used in Section 2.2 for gain and phase margin, yet Equation (2.15) always allows for these type of perturbations. Thus a more stringent requirement on the size of L must be met in Equation (2.15) than that indicated by the special form of L in Equation (2.13). This is another reason (beyond that described in the SISO system example of Section 2.2) that the gain and phase margin predictions in Equations (2.16) and (2.17) are conservative [19]. Thus eigenvalues, which do indicate the sensitivity to a truely diagonal perturbation, are a very important additional tool, although they represent a perturbation that is somewhat too structured.

A more drastic consequence of the unstructured nature of the matrix L is that the singular values of the return difference matrix are not invariant under scale changes [17]. In other words, if the units of the control power variables or the units of the states are changed (and the gains in the feedback loops are changed accordingly, so that the unperturbed closed-loop system characteristics have

not been altered by the scaling), the singular-value plot can change. This fact makes obvious physical sense when one considers cross-feed perturbations: if the units of some of the feedback loops change, and the signals in these loops are allowed to cross-feed to other loops, the boundaries on the allowable crossfeed multiplication factors will change. The upper bound for the size of the L matrix can thus go up and down with scale changes, and the predicted gain and phase margins will change accordingly.

The singular values of a scaled system will always be upper-bounded by the eigenvalues of the return-difference matrix, which are invariant under scale changes [17]. The actual or true gain and phase boundaries as defined by Equation (2.13) are also invariant under scale changes, because they represent perturbations on each loop as it feeds back on itself, and so scales are unimportant. But the singular values will usually underpredict these true margins, because they are accounting for cross-feed disturbances, which can destabilize the system very easily if the units chosen for the system are disadvantageous.

The solution to the problem of the variability of singular values with scaling depends on the control designer's goal. If he or she wishes to study the effects of unmodeled cross-feeds (such as unmodeled coupling between certain states), then he or she must choose units for the system such that all channels will have the same relative magnitudes during the normal operation of the plant. This type of scaling accounts for crossfeeds, but between channels

whose magnitudes are similar. Multiloop gain and phase margins (Equations 2.16 and 2.17) are not particularly meaningful here; the singular values should be interpreted as the allowable 'size' of a fully populated L matrix. On the other hand, if the designer desires to look at multiloop gain and phase margins, which are structured as in Equation (2.13) and do not account for crossfeed perturbations, then he or she should look for the system scaling which yields the largest $\underline{\sigma}[I+HG(j\omega)]_{min}$. Since diagonal perturbations will have identical effects on the scaled and unscaled systems, the boundaries predicted for the 'best possible' scaling can be applied to the unscaled system, and a much less conservative multiloop gain and phase margin will result. It is important to note, however, that boundaries on unstructured L matrices are valid only for the scaling under which they are computed [17]; they cannot be applied to other scalings of the system, because this would involve cross-feeding between feedback paths which have different relative dimensions than those used when the boundaries were computed.

Assuming that the system has been scaled appropriately, the designer can further reduce the conservativeness of singular-value plots by looking at the problem in a subtly different way. If in Figure 2.5, L is replaced by L+I, which represents an additive perturbation, the stability criterion becomes

$$\bar{\sigma}[L(s)] < \underline{\sigma}[I + (HG)^{-1}(s)] , \qquad (2.18)$$

which, when L has the diagonal structure of Equation (2.13), reduces

to the gain and phase boundaries [17]

$$\bar{\sigma}(L) = \sqrt{(1 - k_{max})^2 + 2k_{max}(1 - \cos\phi_{max})}, \qquad (2.19)$$

or

$$GM = 1 \pm \underline{\sigma}_{min}, \qquad (2.20a)$$

$$PM = \pm 2\sin^{-1}(\underline{\sigma}_{min}/2) \text{ if } \underline{\sigma} < 2, \pm 180 \text{ if } \underline{\sigma}_{min} > 2 \qquad (2.20b)$$

Since these boundaries are also sufficient but not necessary (which means they are conservative), the information they yield is complimentary to that obtained from the plot of $\underline{\sigma}[I+HG]$. In other words, it is valid to use the boundary which is least conservative. $[I+(HG)^{-1}]$ is called the inverse return difference matrix, and is easily computed along with the return difference matrix at each frequency. However, the inverse return difference matrix will not be utilized in this paper for several reasons. The first is that the plots of $\underline{\sigma}[I+HG]$ and $\underline{\sigma}[I+(HG)^{-1}]$ tend to be very similar, so that very little new information is generally gained from plotting both. Furthermore, unlike $\underline{\sigma}[I+(HG)^{-1}]$, $\underline{\sigma}[I+HG]$ can be thought of as a Nyquist plot distance. It lends itself to intuitive interpretation much more easily. Finally, the gradients of $\underline{\sigma}[I+(HG)^{-1}]$ are not yet available. Gradients will be introduced in Section 2.4 for $\underline{\sigma}[I+HG]$, and they form an integral part of our sensitivity analysis procedure. For these reasons $\underline{\sigma}[I+HG]$ will be used exclusively throughout the rest of this paper.

The final complication to the singular value analysis is that the location of the disturbance matrix L in the control system is important to the resulting gain and phase margin predictions [18]. This results both from the scaling phenomenon described above, and

from the fact that gain and phase perturbations in different parts of the loop actually have different effects. Unlike SISO systems, in which the location of the disturbances is unimportant, MIMO systems can have cross-feed interactions which cause the locations of phase and gain perturbations to be important. The most important locations for the disturbance matrix L are the plant input and the plant output; these locations are shown in Figure 2.10. Placing L at location 1 in Figure 2.10 yields a robustness measure which is based on the transfer function between R(s) and U(s) in Equation (2.9). This is the measure which we have been discussing up to now. Placing L at location 2 gives a robustness measure based on the following transfer function, between disturbances W(s) and inputs to the control system Z(s),

$$Z(s) = [I + GH(s)]^{-1}W(s). \qquad (2.21)$$

Alternatively, we can simply analyze the system as if the plant is the controller and the controller is the plant. In other words, treat G as H and H as G, and then all the equations for L placed at location 1 will apply to L placed at location 2. The details of the derivations for location 2 will not be discussed, since they are trivial variations on those for location 1. Placing L at these two points can yield very different stability boundaries, and sensitivity information subsequently derived can also be very different. Thus it is important to do the analysis at both points.

At this point, the importance and usefulness of scaling the system appropriately takes on more meaning. At the plant input the

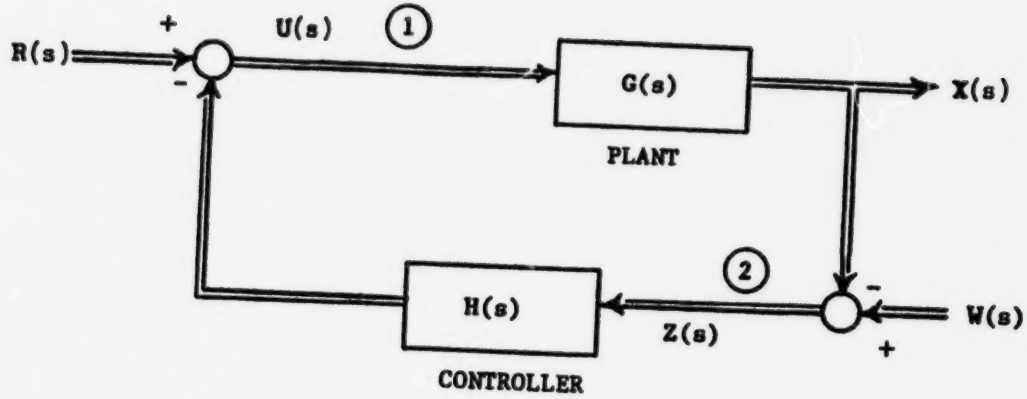


Figure 2.10: Locations for the disturbance matrix, L: 1) input node, 2) output node.

scaling should have the goal of maximizing the singular values, because it is rare for crossfeed perturbations to occur within a control system. Thus the structured or diagonal form of L is the most realistic measure of robustness. At the plant output, however, crossfeeds may occur within the actual plant which were not modeled when the control system was designed. So, to insure a realistic measure of robustness at the output, the scaling used should cause all the states in $\mathbf{x}$ to vary within the same range during the normal operation of the plant.

2.4 SINGULAR-VALUE GRADIENTS

Now that we have a good idea what singular values are and what they mean, we are ready to understand the usefulness of singular-value gradients (also known as $\underline{\sigma}$-gradients). The term "singular-value gradients" was adopted from Reference [8], where the matrix gradients were used in a gradient search design method; here a more appropriate term might be "singular-value partials" because we interpret elements of the matrix gradients as sensitivities, or partials of $\underline{\sigma}$ with respect to the corresponding element of the matrix. A more detailed explanation follows.

The goal of a sensitivity analysis is to identify those parameters, whether they be aerodynamic, control power, or control system parameters, which most greatly impact the stability and/or performance of the closed loop system. Since the minimum singular value, $\underline{\sigma}$, is a measure of relative stability, parameters p for which

$\partial\underline{\sigma}/\partial p$ is large are potentially important parameters. We will describe "how large is large" later. At any rate, if we can compare $\partial\underline{\sigma}/\partial p$ for all the possible "p's" in the system, we can rate the parameters in the order of their importance, taking into account their relative sizes and accuracies. This is the singular-value analysis which we will now develop mathematically.

Consider the state-space representation of the control system of Figure 2.5 to be given by

Plant:

$$\dot{x} = Ax + Bu \tag{2.22}$$

$$z = Tx \tag{2.23}$$

Control Law:

$$u = -Kz + r \tag{2.24}$$

Equation (2.22) represents a plant of order N_s having N_o output measurements, z, modeled by Equation (2.23), and N_c control inputs, u. Equation (2.24) represents the feedback control law driven by the sensor output, z, and reference input signal, r. In terms of transfer matrices (taking Laplace transforms), the control law is given by

$$U(s) = -K[T(Is - A)^{-1}B]U(s) + R(s) \tag{2.25}$$

Therefore, the control input can be written as

$$U(s) = [I + HG(s)]^{-1}R(s) \tag{2.26}$$

where $H(s) = K$ and $G(s) = [T(Is-A)^{-1}B]$.

Equations (2.22) to (2.24) can be written in an augmented form as

$$\dot{x} = \bar{A}x + \bar{B}u \ , \tag{2.27}$$

$$u = -\bar{C}x + r \ , \tag{2.28}$$

where $\bar{A} = A$, $\bar{B} = B$, and $\bar{C} = KT$ (this notation is used to conform to the notation in Reference [3]). Then the Laplace transform of the control input can be expressed as follows:

$$U(s) = [I + \bar{C}(Is - \bar{A})^{-1}\bar{B}]^{-1}R(s) \ , \tag{2.29}$$

and, therefore, the return difference matrix, (I+HG), can be represented as

$$(I + HG) = [I + \bar{C}(Is - \bar{A})^{-1}\bar{B}]. \tag{2.30}$$

In the case in which the control law includes controller dynamics, a similar derivation would be involved [3], but the computation of $\bar{A}$, $\bar{B}$, and $\bar{C}$, would be different. The $\bar{A}$, $\bar{B}$, and $\bar{C}$ matrices can also be found for the case where the loop is broken at the plant output. The resulting transfer function is between disturbances at the sensors [W(s) in Figure 2.10] and the output vector, Z(s). The analysis then proceeds exactly as follows, with the realization that the elements of $\bar{A}$, $\bar{B}$, and $\bar{C}$ may be different than those for the input case.

The singular values of (I+HG) are $\sigma_i(s)$, and the corresponding right and left normalized singular vectors are $v_i(s)$ and $u_i(s)$, respectively (these are simply columns of the matrices U and V which result from the decomposition of (I+HG) into USV^*). Hence by definition

$$(I + HG)v_i = u_i\sigma_i \tag{2.31}$$

$$(I + HG)^* u_i = v_i \sigma_i \tag{2.32}$$

for i = 1, 2, ... N_c, where W^* means the conjugate-transpose of W.

The normalized eigenvectors satisfy the following orthogonal properties:

$$u_i^* v_j = \delta_{ij} \text{ and } v_i^* v_j = \delta_{ij}, \tag{2.33}$$

where δ_{ij} is the Kroneker delta which is unity when i=j and zero when $i \neq j$.

Let p be a parameter for which sensitivity information is needed. Differentating Equation (2.31) and (2.32) with respect to p and then premultiplying the result by u_i^* and v_i^*, respectively, and adding them together, one obtains

$$u_i^* \frac{\partial(I + HG)}{\partial p} v_i + v_i^* \frac{\partial(I + HG)^*}{\partial p} u_i + [u_i^*(I + HG) - v_i^* \sigma_i] \frac{\partial v_i}{\partial p} +$$

$$+ (v_i^*(I + HG)^* - u_i^* \sigma_i) \frac{\partial u_i}{\partial p} = \frac{\partial \sigma_i}{\partial p} (u_i^* u_i + v_i^* v_i) \tag{2.34}$$

Using Equation (2.31) to (2.33) in (2.34), one obtains

$$\frac{\partial \sigma_i}{\partial p} = \frac{1}{2} \left(u_i^* \frac{\partial(I + HG)}{\partial p} v_i + v_i^* \frac{\partial(I + HG)^*}{\partial p} u_i\right) \tag{2.35}$$

Notice that the first and second terms in the right-hand side of the equation are complex conjugates. Therefore, Equation (2.35) can be written as

$$\frac{\partial \sigma_i}{\partial p} = \text{Real part of } \left[u_i^* \frac{\partial(I + HG)}{\partial p} v_i\right]$$

$$= \text{Re}\left[u_i^* \frac{\partial(I + HG)}{\partial p} v_i\right] \tag{2.36}$$

Using Equation (2.30) and letting

$$\Phi = (Is - \bar{A})^{-1}, \tag{2.37}$$

Equation (2.36) can be written as

$$\frac{\partial \sigma_i(I + HG)}{\partial p} = Re \cdot tr\left[\frac{\partial(I + \bar{C}\Phi\bar{B})}{\partial p} v_i u_i^*\right] \tag{2.38}$$

where tr[W] means the trace of W and is equal to the sum of the elements in the principal diagonal of W. Equation (2.38) can be expanded as

$$\frac{\partial \sigma_i}{\partial p} = Re \cdot tr\left[\{\bar{C}\Phi \frac{\partial \bar{A}}{\partial p} \Phi\bar{B} + \bar{C}\Phi \frac{\partial \bar{B}}{\partial p} + \frac{\partial \bar{C}}{\partial p} \Phi\bar{B}\} v_i u_i^*\right] \tag{2.39}$$

It is now possible to obtain three expressions for $\partial\sigma_1/\partial p$, one for each of the following cases:

1) The parameter p is an element of $\bar{A}$, i.e., $p = p_{\bar{A}}$

2) The parameter p is an element of $\bar{B}$, i.e., $p = p_{\bar{B}}$

3) The parameter p is an element of $\bar{C}$, i.e., $p = p_{\bar{C}}$.

The resulting expressions are

$$\frac{\partial \sigma_i}{\partial p_{\bar{A}}} = Re \cdot tr[\bar{C}\Phi \frac{\partial \bar{A}}{\partial p_{\bar{A}}} \Phi\bar{B} v_i u_i^*] \tag{2.40}$$

$$\frac{\partial \sigma_i}{\partial p_{\bar{B}}} = Re \cdot tr[\bar{C}\Phi \frac{\partial \bar{B}}{\partial p_{\bar{B}}} v_i u_i^*] \tag{2.41}$$

$$\frac{\partial \sigma_i}{\partial p_{\bar{C}}} = Re \cdot tr[\frac{\partial \bar{C}}{\partial p_{\bar{C}}} \Phi\bar{B} v_i u_i^*] \tag{2.42}$$

By recalling the following matrix operation [12],

$$\frac{\partial}{\partial X}\,[\mathrm{tr}\{YXZ\}] = Y^{*}Z^{*} \tag{2.43}$$

(where $\partial/\partial X$ indicates element-by-element derivatives),

and the matrix trace property [8],

$$\mathrm{Re}\cdot\mathrm{tr}(A) = \mathrm{Re}\cdot\mathrm{tr}(A^{*}) \tag{2.44}$$

it is possible to extend Equations (2.40) through (2.42) to matrix form. For $\bar{A}$,

$$\frac{\partial\sigma_i}{\partial\bar{A}} = \frac{\partial}{\partial\bar{A}}\,[\mathrm{Re}\cdot\mathrm{tr}\{\bar{C}\Phi\bar{A}\Phi\bar{B}(v_i u_i^{*})\}]$$

$$= \mathrm{Re}[(\bar{C}\Phi)^{*}(\Phi\bar{B}v_i u_i^{*})^{*}] \tag{2.45}$$

where $\partial\sigma_i/\partial\bar{A}$ is a matrix whose elements are $\partial\sigma_i/\partial a_{ij}$. Using the transpose of $\bar{A}$ simplifies this expression further.

$$\frac{\partial\sigma_i}{\partial\bar{A}^{T}} = \mathrm{Re}[\Phi\bar{B}v_i u_i^{*}\bar{C}\Phi] \tag{2.46}$$

Similarly, for $\bar{B}$,

$$\frac{\partial\sigma_i}{\partial\bar{B}} = \mathrm{Re}[(\bar{C}\Phi)^{*}(v_i u_i^{*})^{*}] \tag{2.47}$$

or using the transpose,

$$\frac{\partial\sigma_i}{\partial\bar{B}^{T}} = \mathrm{Re}[(v_i u_i^{*})\bar{C}\Phi] \tag{2.48}$$

and for matrix $\bar{C}$,

$$\frac{\partial\sigma_i}{\partial\bar{C}} = \mathrm{Re}[I^{T}(\Phi\bar{B}(v_i u_i^{*})^{*}] \tag{2.49}$$

or using the transpose of $\bar{C}$,

$$\frac{\partial \sigma_i}{\partial \bar{C}^T} = \mathrm{Re}(\Phi \bar{B} v_i u_i^*) \ . \tag{2.50}$$

Expressions (2.46), (2.48), and (2.50) can be used to evaluate the singular-value gradients with respect to elements of the system and controller matrices. Note that the gradients, like the singular values, are functions of frequency; thus singular-value-gradient plots (or $\underline{\sigma}$-gradient plots) can be obtained over a range of frequencies for <u>each element of interest</u>. Note also that the information necessary to obtain the gradients is already available from the calculation of the $\underline{\sigma}$-plot (if Φ is evaluated explicitly). This is because v_i and u_i are products of the calculation of the singular values. Therefore, if one is computing the $\underline{\sigma}$-plot, and Φ is directly available, little additional computational effort is needed to calculate the $\underline{\sigma}$-gradient plots.

To determine the frequency at which a particular $\underline{\sigma}$-gradient plot is most important, we ask the following question: What is the smallest <u>percentage</u> change in p necessary to drive the singular-value plot to an undesirably low value? If we choose the "undesirably low value" to be .2 (which translates into a GM of -1.6 to 1.9 dB and a PM of ±11.5 degrees), this question can be written mathematically as

$$\min_{0<\omega<\infty} \left[\frac{\Delta p}{p}\right] = \min_{0<\omega<\infty} \left[\frac{\underline{\sigma} - \underline{\sigma}_{MA}}{\frac{\partial \underline{\sigma}}{(\partial p/p)}} \right] \tag{2.51}$$

Where the "normalized" singular-value gradient has been employed to account for differences in the size of different elements. For each parameter there is a frequency for which the above expression is minimized.

Graphically, Equation (2.51) finds the frequency and $\Delta p/p$ for which the situation illustrated in Figure 2.11 occurs. 2.11(a) shows the original system $\underline{\sigma}$-plot, and 2.11(b) shows a sample normalized gradient curve for the same system. If the $\Delta p/p$ found from Equation (2.51) is introduced, the resulting perturbed-system $\underline{\sigma}$-plot is shown in Figure 2.11(c). The frequency where Equation (2.51) is minimized is the point where the perturbed plot touches the $\underline{\sigma}$=.2 line in Figure 2.11(c). Note that this frequency does not necessarily correspond to the frequency of the minimum of the original $\underline{\sigma}$-plot.

The SVA approach is to compile a table of parameters, indicating for each one the minimum percentage change it must undergo to drive the system nearly unstable. This table is a first step in determining the parameters that are of greatest importance. The other steps to be taken are

1) Determine whether the frequency at which the parameter effects the singular value is a critical one. If the $\Delta p/p$ required is relatively small but not extremely small, and the frequency of its effect is either very low (such as a parameter which excites a spiral instability) or very high (such as a structural mode which the pilot will not be able to notice), the parameter may be judged to be relatively unimportant.

2) Obtain an estimate of the accuracy of the parameter. If the system is extremely sensitive to a certain parameter, but that parameter is known to a high degree of accuracy, then the parameter is not important.

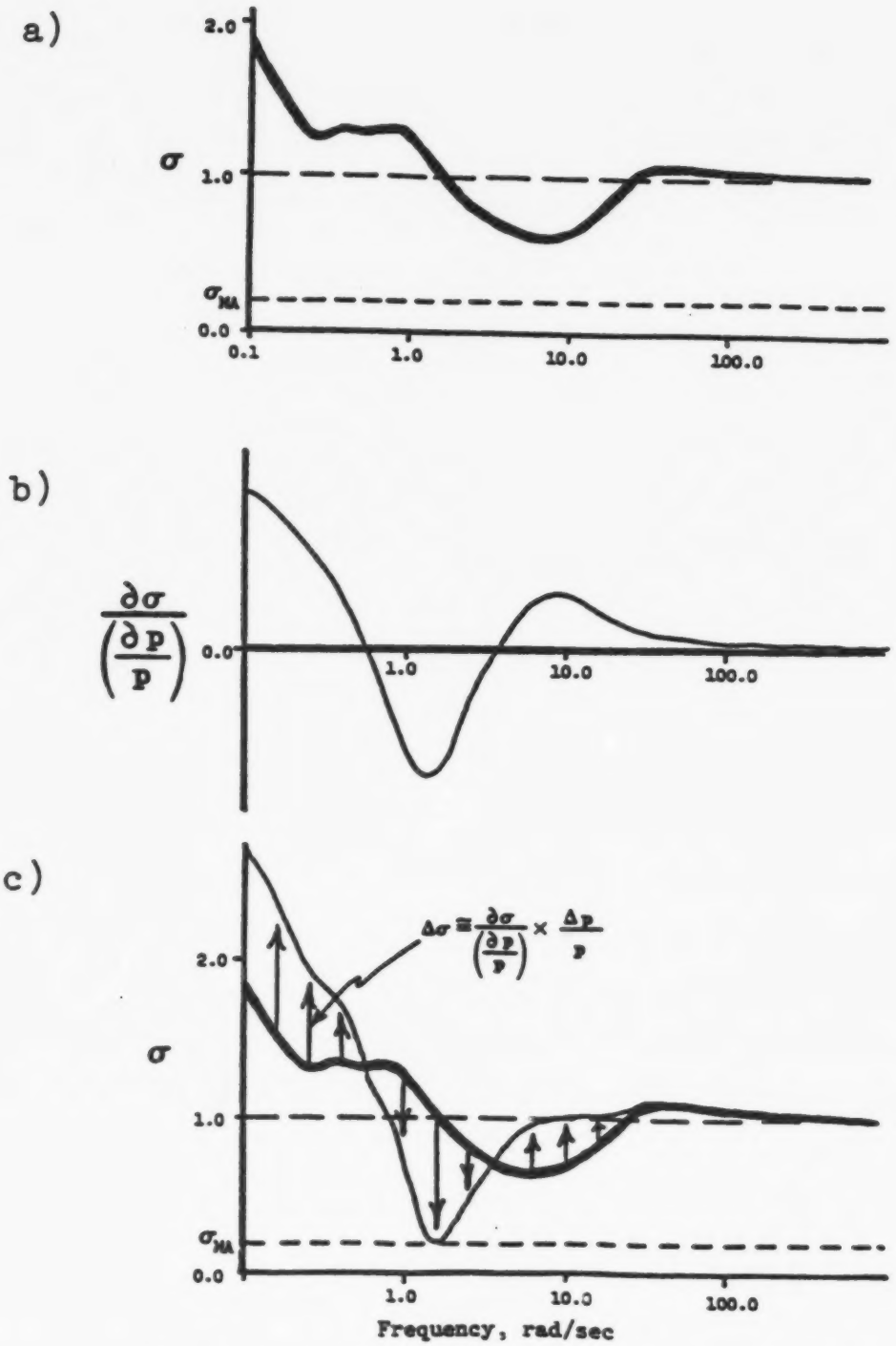


Figure 2.11: Effect of a parameter variation on the σ-plot, as predicted by the σ-gradient

3) Determine whether the parameter can vary independent of other parameters. This is an extremely important point, because elements of a block diagram which has been augmented to an aircraft often make up several elements of the $\bar{A}$ or $\bar{B}$ matrix, which must change together. This problem is dealt with in Section 4.

4) Check to make sure that the linearity assumption made in the first-order approximation shown in Figure 2.11 is valid. If $\bar{\sigma}$ must change by more than about 45% in order to reach the minimum allowable level, than this approximation is generally not very good [23].

5) Check the sizes of the other singular values (that is, those that are not the minimum), and of their gradients. If some of the other singular values are relatively small and/or there gradients are large, they must be included in the tabulation of sensitivities.

6) Perform the SVA and steps 1) through 5) for the system with the loop broken at the plant output.

Only after all of the above steps have been taken can a good picture of the sensitivity to a parameter be determined.

Figure 2.12 recaps the SVA. It must be conducted for the loop broken at both the plant input and the plant output. First, the $\underline{\sigma}$-plot of the unperturbed system is calculated, along with the closed-loop system eigenvalues. System scaling should be optimized at this point, and the scaling chosen should then be used for the rest of the analysis. The $\underline{\sigma}$-plot will tell the designer the minimum singular value, and therefore the phase and gain margins of the unperturbed system, using the universal diagram (Figure 2.6) or Equations (2.17a) and (2.17b). The closed loop eigenvalues must be calculated to determine whether the system is stable or not. It must be stressed that the closed-loop eigenvalues are not to be confused with the eigenvalues of the return difference matrix, which are plotted against frequency like the singular values. The second

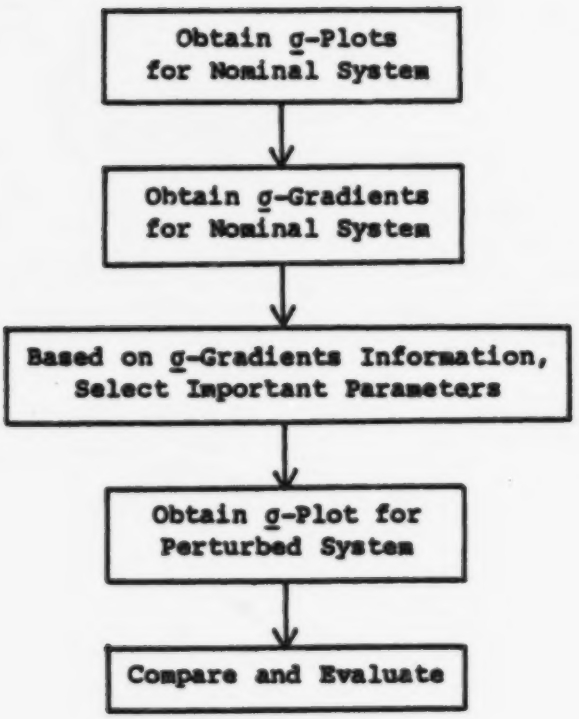


Figure 2.12: Possible flow chart of singular value Sensitivity analysis technique

step in the sensitivity analysis is to evaluate the singular-value gradients ($\underline{\sigma}$-gradient plots) for the unperturbed system. These gradient plots can then be reduced to a table of minimum parameter-variations required to drive the system to some predetermined level of relative stability. This "minimum allowable" level of relative stability is represented by a minimum allowable value of the $\underline{\sigma}$-plot. The next step is to evaluate the $\underline{\sigma}$-plot, closed-loop system eigenvalues, and any other desired performance measures, for the system after it has been perturbed by variations in selected parameters. This final step adds confidence to the analysis.

3. COMPUTING GRADIENTS FOR DIGITAL CONTROL SYSTEMS

The singular-value-gradient equations derived in Section 2 are for the following terms: $\partial\underline{\sigma}/\partial\bar{A}^T$, $\partial\underline{\sigma}/\partial\bar{B}^T$, and $\partial\underline{\sigma}/\partial\bar{C}^T$. We will see in the following section that these equations are valid for digital systems if A and B are replaced with ϕ and Γ, the transition and discrete control power matrices, respectively. But the elements of these matrices have no physical significance; they no longer represent simple linear combinations of aerodynamic and control power derivatives. In fact, the gradients with respect to ϕ and Γ, $\partial\sigma/\partial\phi^T$ and $\partial\underline{\sigma}/\partial\Gamma^T$, have no real value by themselves, because none of the terms of ϕ and Γ can vary independently. What one would really like to have are the derivatives with respect to the matrices before discretization. This requires a new set of equations. These equations are derived in this section, using extensions of the theory presented in Section 2.

3.1 THE EFFECT OF DISCRETIZATION ON THE SVA

When a single-rate digital control system is designed, the continuous plant (including all servo, sensor, and analog control dynamics) must be discretized. The system

$$\dot{x} = A_c x + B_c u \tag{3.1}$$

is often discretized using the following equations [16]:

$$\phi(A_c,T) = I + A_c T + \frac{A_c^2 T^2}{2!} + \frac{A_c^3 T^3}{3!} + \ldots, \text{ and} \tag{3.2}$$

$$\Gamma(B_c,T) = \int_0^T \phi(A_c,\tau)d\tau \cdot B_c \ , \tag{3.3}$$

where

T = sample time.

The discrete representation of the system is then

$$x_{k+1} = \phi x_k + \Gamma u_k. \tag{3.4}$$

This discretization is performed so that analysis and augmentation can be performed in one domain, namely the discrete domain. Feedback gains and digital dynamics (such as digital filters and compensators) are then augmented to the discretized model to form the final form of the control system,

$$x_{k+1} = \left[\begin{array}{c|c} \phi(A_c,T) & L12 \\ \hline L21 & L22 \end{array}\right] x_k + \left[\begin{array}{c} \Gamma(B_c,T) \\ \hline G21 \end{array}\right] u_k, \tag{3.5}$$

$$u_k = -\bar{C}\, x_k \tag{3.6}$$

(Output, or observer, equations are often used in the analysis and design of these systems; these equations can be reduced to the form shown above.) Singular value plots can be found for control systems of this form, as outlined by Broussard [13]. However, the direct application of the technique for finding singular-value gradients presented by Herrera [1] and Newsom [8] would find gradients with respect to the elements of the augmented discrete-system matrices in Equations (3.5) and (3.6). For parameter sensitivities, we require gradients with respect to the elements of the A_c and B_c matrices. The following section derives these gradients, beginning the derivation with Newsom's results. No new equations are necessary

for the $\bar{C}$ matrix gradients, because the gain matrix, which represents the discrete controller, exists in the real world, and so gradients with respect to the $\bar{C}$ matrix are valid. For this reason the notation for the feedback gain matrix has not been changed. It should also be noted that for all quadrants of the partitioned matrices in Equation (3.5) except the upper left-hand corner, the gradients are still valid. They represent the gradients with respect to parameters in the digital control system, such as digital filters and compensators.

3.2 DERIVATION OF THE SINGULAR-VALUE GRADIENTS FOR DIGITAL SYSTEMS

Because the continuous and digital parts of the control system are inherently different, the technique of switching their roles to obtain singular values and their gradients for the case where disturbances are measured at the output (location 2 in Figure 2.10) instead of the input (location 1 in Figure 2.10) is not valid. The nature of the continuous-to-digital and the digital-to-continuous interconnections must be accounted for. Therefore, in this section, we present the derivation of the $\underline{\sigma}$-gradients for digital systems in two parts. Section 3.2.1 discusses the $\underline{\sigma}$-gradients for the disturbance located at the plant input, and Section 3.2.2 discusses the sigma-gradients for the disturbance located at the plant output.

3.2.1 Digital Gradients for the Input Case

For now we will assume that no digital dynamics have been added to the digitized plant (this restriction will be lifted later). The system, then, is simply

$$x_{k+1} = \phi x_k + \Gamma u_k \tag{3.7}$$

$$u_k = -\bar{C} x_k \ , \tag{3.8}$$

and Newsom's results (Equation 2.39) are valid for the gradients with respect to <u>any</u> parameter (here we use a_c to denote an element of the continuous A_c matrix):

$$\frac{\partial \sigma_i}{\partial a_c} = \mathrm{Re} \cdot \mathrm{tr}[(\bar{C}\Omega \frac{\partial \phi}{\partial a_c} \Omega\Gamma + \bar{C}\Omega \frac{\partial \Gamma}{\partial a_c}) v_i u_i^{*}] \ , \tag{3.9}$$

where: $\Omega = (Iz - \phi)^{-1}$ and $z = e^{-sT}$ [13].

(Here we have simply replaced $\bar{A}$ with ϕ and $\bar{B}$ with Γ in Equation (2.39), and made the adjustments necessary when analyzing digital systems.) The first step is to evaluate $\partial\phi/\partial a_c$ and $\partial\Gamma/\partial a_c$. The equations below are approximate, and are taken from Maine and Iliff, Reference [14]. Appendix A gives derivations for these equations.

$$\partial\phi/\partial a_c = \frac{1}{2} \Psi(\frac{\partial A_c}{\partial a_c})(\phi + I) \ , \quad \text{where } \Psi = \int_0^T \phi(A_c, \tau) d\tau; \tag{3.10}$$

$$\partial\Gamma/\partial a_c = \frac{1}{2} \Psi(\frac{\partial A_c}{\partial a_c}) \Psi B_c . \tag{3.11}$$

These approximations are exact to order T^2 [14]. Substituting these expressions back into the original equations and using the identity $\Gamma = \Psi B_c$ yields

$$\frac{\partial \sigma_i}{\partial a_c} = \mathrm{Re}\cdot\mathrm{tr}\{\tfrac{1}{2}\,[\bar{C}\Omega\Psi(\frac{\partial A_c}{\partial a_c})(\phi + I)\Omega\Gamma + \bar{C}\Omega\Psi(\frac{\partial A_c}{\partial a_c})\Gamma]\Psi_i u_i^{*}\}\ . \tag{3.12}$$

To this expression we apply the following simplified application of Equations (2.43) and (2.44):

if $$\partial p/\partial a = \mathrm{Re}\cdot\mathrm{tr}\,[Y(\partial A/\partial a)Z]\ , \tag{3.13a}$$

then

$$\partial p/\partial A^T = YZ\ . \tag{3.13b}$$

The matrix expression for the A_c matrix gradient, according to this rule, is

$$\frac{\partial \sigma_i}{\partial A_c^T} = \frac{1}{2}\,[(\phi + I)\Omega\Gamma\Psi_i u_i^{*}\bar{C}\Omega\Psi + \Gamma\Psi_i u_i^{*}\bar{C}\Omega\Psi]. \tag{3.14}$$

At this point we recognize the following formulas from Section 2.3:

$$\frac{\partial \sigma_i}{\partial \phi} = \mathrm{Re}[\Omega\Gamma\Psi_i u_i^{*}\bar{C}\Omega], \tag{3.15}$$

and

$$\frac{\partial \sigma_i}{\partial \Gamma} = \mathrm{Re}[\Psi_i u_i^{*}\bar{C}\Omega] \tag{3.16}$$

(where $\bar{A}$ and $\bar{B}$ have been replaced by ϕ and Γ respectively, and Ω is as defined in Equation (3.9)). These equations allow Equation (3.14) to be further

simplified to

$$\frac{\partial \sigma_i}{\partial A_c^T} = \frac{1}{2}\,[(\phi + I)\,\frac{\partial \sigma_i}{\partial \phi^T} + \Gamma\,\frac{\partial \sigma_i}{\partial \Gamma^T}]\Psi. \tag{3.17}$$

The derivation of $\partial \sigma_i/\partial b_c$ where b_c is an element of B_c and the control system is digital is much simpler. Starting, again, with Mukhopadhyay's Equation (2.39) where $\bar{B}$ is replaced by Γ and terms that are zero have been left out:

$$\frac{\partial \sigma_i}{\partial b_c} = \bar{C}\Omega \frac{\partial \Gamma}{\partial b_c} v_i u_i^* \tag{3.18}$$

we first use the fact that $\Gamma = \Psi B_c$ to get

$$\frac{\partial \sigma_i}{\partial b_c} = \bar{C}\Omega \frac{\partial (\Psi B_c)}{\partial b_c} v_i u_i^* \; ; \tag{3.19}$$

and, since Ω is not a function of B_c,

$$\frac{\partial \sigma_i}{\partial b_c} = \bar{C}\Omega\Psi \frac{\partial B_c}{\partial b_c} v_i u_i^* . \tag{3.20}$$

Finally, using (3.13) and (3.16)

$$\frac{\partial \sigma_i}{\partial B_c^T} = v_i u_i^* \bar{C}\Omega\Psi; \tag{3.21}$$

$$\frac{\partial \sigma_i}{\partial B_c^T} = \frac{\partial \sigma_i}{\partial \Gamma^T} \Psi . \tag{3.22}$$

It remains to show that these equations can somehow be applied to the matrices in Equations (3.5) and (3.6), which have been augmented in the digital domain. To do this, first assume that the system has been modeled as shown in Figure 3.1, where none of the digital-to-continuous or continuous-to-digital connections have been made, but all dynamics are completely modeled. The discretized continuous plant dynamics have the form

$$x_{k+1} = \phi x_k + \Gamma u_k \tag{3.23}$$

$$y_k = H x_k \tag{3.24}$$

and the digital control system dynamics have the form

$$x_{d_{k+1}} = A_d x_{d_k} + B_d u_{d_k} \tag{3.25}$$

$$y_{d_k} = H_d x_d + F_d u_{d_k} \; . \tag{3.26}$$

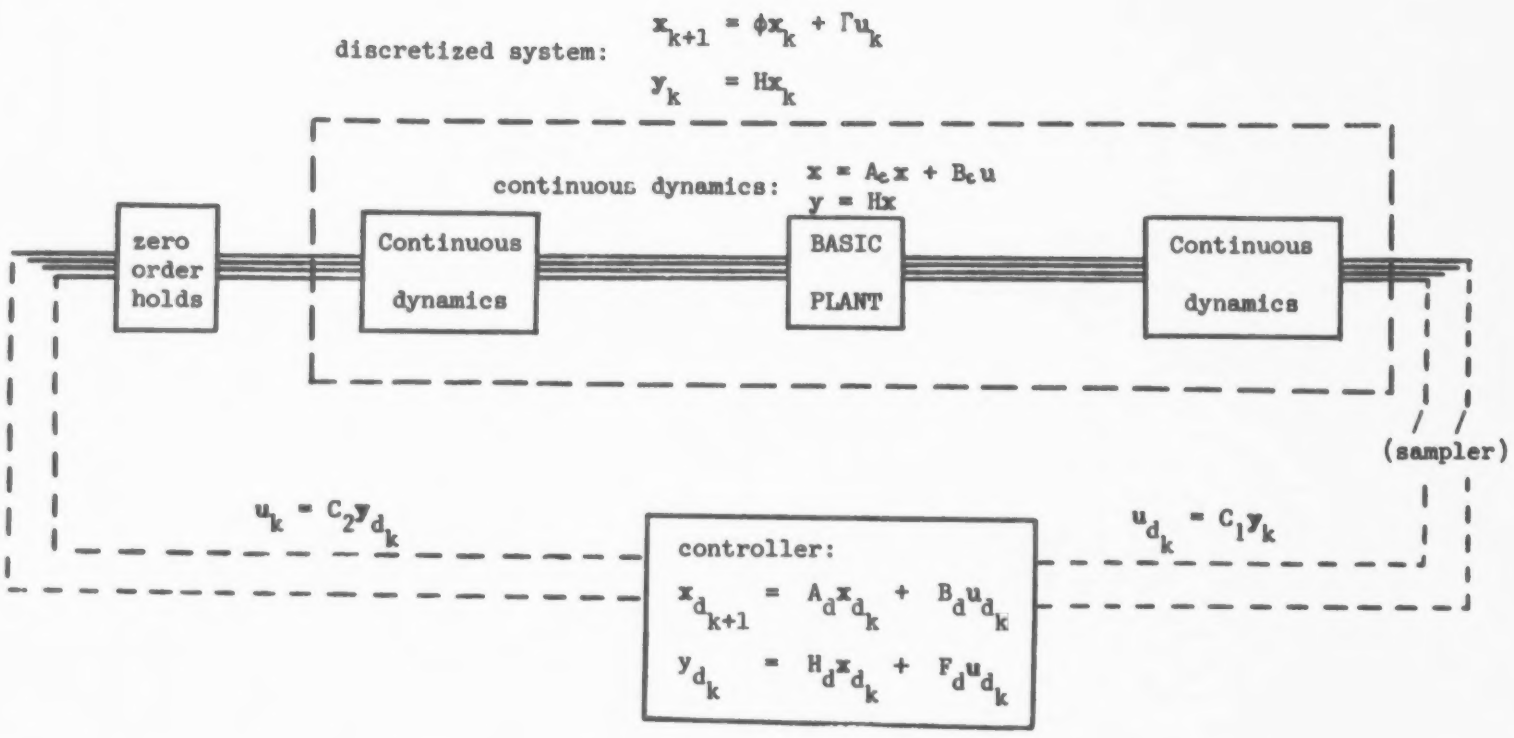

discretized system: $x_{k+1} = \phi x_k + \Gamma u_k$
$y_k = Hx_k$
continuous dynamics: $\dot{x} = A_c x + B_c u$
$y = Hx$
zero order holds
Continuous dynamics
BASIC PLANT
Continuous dynamics
(sampler)
$u_k = C_2 y_{d_k}$
$u_{d_k} = C_1 y_k$
controller:
$x_{d_{k+1}} = A_d x_{d_k} + B_d u_{d_k}$
$y_{d_k} = H_d x_{d_k} + F_d u_{d_k}$

Figure 3.1: Allowable control system structure for singular value gradients of digital systems

This system can be represented by the completely decoupled set of matrix equations

$$\left[\begin{array}{c|c} x_{k+1} \\ \hline x_{d_{k+1}} \end{array}\right] = \left[\begin{array}{c|c} \phi(A_c,T) & 0 \\ \hline 0 & A_d \end{array}\right] \left[\begin{array}{c|c} x_k \\ \hline x_{d_k} \end{array}\right] + \left[\begin{array}{c|c} \Gamma(B_c,T) & 0 \\ \hline 0 & B_d \end{array}\right] \left[\begin{array}{c|c} u_k \\ \hline u_{d_k} \end{array}\right] \tag{3.27}$$

$$\left[\begin{array}{c|c} y_k \\ \hline y_{d_k} \end{array}\right] = \left[\begin{array}{c|c} H & 0 \\ \hline 0 & H_d \end{array}\right] \left[\begin{array}{c|c} x_k \\ \hline x_{d_k} \end{array}\right] + \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & F_d \end{array}\right] \left[\begin{array}{c|c} u_k \\ \hline u_{d_k} \end{array}\right] . \tag{3.28}$$

Since this section deals with the disturbance located at the input to the system, the desired transfer function is between the closed-loop system input and the input to the plant. To obtain this transfer function, we make the continuous-to-digital connections.

$$\begin{aligned} u_d &= C_1 y_k \\ &= C_1 H x_k . \end{aligned} \tag{3.29}$$

This allows u_d to be deleted from the augmented vector $[u_k^T \mid u_{dk}^T]^T$, and y_k to be deleted from augmented vector $[y_k^T \mid y_{dk}^T]^T$. These vectors have been internalized into the system by Equation (3.29). Equations (3.27) and (3.28) now have the form

$$\left[\begin{array}{c|c} x_k \\ \hline x_{d_k} \end{array}\right] = \left[\begin{array}{c|c} \phi & 0 \\ \hline B_d C_1 H & A_d \end{array}\right] \left[\begin{array}{c|c} x_k \\ \hline x_{d_k} \end{array}\right] + \left[\begin{array}{c} \Gamma \\ \hline 0 \end{array}\right] u_k \quad ; \tag{3.30}$$

$$y_d = \left[\begin{array}{c|c} F_d C_1 H & H_d \end{array}\right] \left[\begin{array}{c|c} x_k \\ \hline x_{d_k} \end{array}\right] . \tag{3.31}$$

If we now let $u_k = C_2 y_{d_k}$, then this system is in exactly the form presented in Equations (3.5) and (3.6), with the following matrix as $\bar{C}$:

$$\bar{C} = -C_2[\ F_d C_1 H \mid H_d\ ]. \tag{3.32}$$

If the matrices in Equations (3.30) and (3.32) are taken as the $\bar{A}$, $\bar{B}$, and $\bar{C}$ matrices describing the system, then it is apparent that

$$\frac{\partial \bar{A}}{\partial a_c} = \left[\begin{array}{c|c} \partial\phi/\partial a_c & 0 \\ \hline 0 & 0 \end{array}\right], \text{ and} \tag{3.33}$$

$$\frac{\partial \bar{B}}{\partial a_c} = \left[\begin{array}{c} \partial\Gamma/\partial a_c \\ \hline 0 \end{array}\right]. \tag{3.34}$$

This allows Equations (3.9) and (3.14) to be written as

$$\frac{\partial \sigma_i}{\partial a_c} = \mathrm{Re}\cdot\mathrm{tr}\ \{[(\bar{C}\Omega)_1 \frac{\partial \phi}{\partial a_c} (\Omega\bar{B})_1 + (\bar{C}\Omega) \frac{\partial \Gamma}{\partial a_c}]\Psi_i u_i\}, \text{ and },$$

$$\frac{\partial \sigma_i}{\partial A_c^T} = \frac{1}{2}\,\mathrm{Re}\{[(\phi + I)(\Omega\bar{B})_1\ \Psi_i u_i^*(\bar{C}\Omega)_1 + \Gamma\Psi_i u_i^*(\bar{C}\Omega)_1]\Psi\} \tag{3.35}$$

where the subscripted notation indicates that $(\bar{C}\Omega)$, and $(\Omega\bar{B})$ in Equation (3.35) have been partitioned appropriately.

Finally, we recognize that the gradients $\partial\sigma_i/\partial\bar{A}^T$ and $\partial\sigma_i/\partial\bar{B}^T$ can be written as follows:

$$\frac{\partial \sigma_i}{\partial \bar{A}^T} = \Omega\bar{B}\Psi_i u_i^*\bar{C}\Omega = \left[\frac{(\Omega\bar{B})_1}{(\Omega\bar{B})_2}\right]^* \Psi_i u_i^*[(\bar{C}\Omega)_1 | (\bar{C}\Omega)_2]$$

$$-\left[\begin{array}{c|c} (\Omega\bar{B})_1\Psi_i u_i^*(\bar{C}\Omega)_1 & (\Omega\bar{B})_1\Psi_i u_i^*(\bar{C}\Omega)_2 \\ \hline (\Omega\bar{B})_2\Psi_i u_i^*(\bar{C}\Omega)_1 & (\Omega\bar{B})_2\Psi_i u_i^*(\bar{C}\Omega)_2 \end{array}\right] \tag{3.36}$$

$$\frac{\partial \sigma_i}{\partial \bar{B}^T} = v_i u_i^* \bar{C}\Omega = v_i u_i^* [(\bar{C}\Omega)_1 \mid (\bar{C}\Omega)_2] = [v_i u_i^* (\bar{C}\Omega)_1 \mid v_i u_i^* (\bar{C}\Omega)_2] , \tag{3.37}$$

which means that the partitions in Equation (3.35) can be obtained directly from partitions of $\partial \sigma_i / \partial \bar{A}^T$ and $\partial \sigma_i / \partial \bar{B}^T$. Combining Equations (3.35), (3.36) and (3.37) brings us to the conclusion that Equations (3.17) and (3.22) are valid in the upper left-hand quadrant of $\bar{A}$ and the upper quadrant of $\bar{B}$ in Equation 3.30 if they are written with the proper partitions of $\partial \sigma_i / \partial \bar{A}^T$ and $\partial \sigma_i / \partial \bar{B}^T$.

$$\frac{\partial \sigma}{\partial A_c^T} = \mathrm{Re}\left\{\frac{1}{2} \left[(\phi + I)\left(\frac{\partial \sigma}{\partial \bar{A}^T}\right)_{11} + \bar{B}_1 \left(\frac{\partial \sigma}{\partial \bar{B}^T}\right)_1\right]\psi\right\} \tag{3.38}$$

Note that the following assumptions have been made:

(a) All feedbacks (connections between the digital and the continuous system) have been made through zero-order holds. That is why the Γ matrix appears in the upper quadrant of Equations (3.27) and (3.30).

(b) The gain matrix $\bar{C}$ is as defined in Equations (3.32) and as shown in Figure (3.1). This means that continuous feedbacks must be defined before the continuous system is discretized, and no feedbacks can be defined from digital blocks to other digital blocks: they must go from the continuous to the digital system.

(c) The continuous system in Equations (3.23) and (3.24) have no direct link; that is, it has no u terms in the output equation. This is not a restrictive assumption for most real systems.

3.2.2 Digital Gradients for the Output Case

Gradients for the case where the disturbance matrix L is placed at the output of the plant are obtained by starting with Equations (3.27) and (3.28), and making the digital-to-continuous connections instead of the continuous-to-digital connections. This is done with the equation

$$u_k = C_2 y_{dk}$$

$$= C_2 H_d x_{dk} + C_2 F_d u_{dk} \tag{3.39}$$

which causes u_k and y_{dk} to drop out of Equations (3.27) and (3.28), leaving the system equations

$$\begin{bmatrix} x_k \\ x_{d_k} \end{bmatrix} = \left[\begin{array}{c|c} \phi & \Gamma C_2 H_d \\ \hline 0 & A_d \end{array}\right] \begin{bmatrix} x_k \\ x_{d_k} \end{bmatrix} + \left[\begin{array}{c} \Gamma C_2 F_d \\ \hline B_d \end{array}\right] u_k; \tag{3.40}$$

$$y_k = [\, H \mid 0 \,].$$

If the matrices in Equation (3.40) are taken to be $\bar{A}$, and $\bar{B}$, while $\bar{C}$ is found by letting $u_{dk} = C_1 y_k$:

$$\bar{C} = -C_1[\, H \mid 0 \,], \tag{3.41}$$

then the system takes the form necessary to do singular values and their gradients for the case where the perturbation matrix L is placed at the plant output. For this case the derivatives of $\bar{A}$ and $\bar{B}$ take on a slightly different form:

$$\frac{\partial \bar{A}}{\partial a_c} = \left[\begin{array}{c|c} \partial\phi/\partial a_c & [\partial\Gamma/\partial a_c C_2 H_d) \\ \hline 0 & 0 \end{array}\right], \text{ and} \tag{3.42}$$

$$\frac{\partial \bar{B}}{\partial a_c} = \left[\begin{array}{c} (\partial\Gamma/\partial a_c) C_2 F_d \\ \hline 0 \end{array}\right]. \tag{3.43}$$

Plugging these results into Equation (2.39) and using the proper partitions of the matrices gives

$$\frac{\partial \sigma_i}{\partial a_c} = \text{Re} \cdot \text{tr} \left\{ \left(\left[(\bar{C}\Omega)_1 \frac{\partial \phi}{\partial a_c} \;\middle|\; (\bar{C}\Omega)_1 \frac{\partial \Psi}{\partial a_c} B_c C_2 H_d \right] \frac{(\Omega\bar{B})_1}{(\Omega\bar{B})_2} \right.\right.$$

$$\left.\left. + (\bar{C}\Omega)_1 \frac{\partial \Psi}{\partial a_c} B_c C_2 F_d \right) \Psi_1 u_1^* \right\}, \tag{3.44}$$

which is easily reduced to

$$\frac{\partial \sigma_i}{\partial a_c} = \mathrm{Re} \cdot \mathrm{tr} \{[(\bar{C}\Omega)_1 \frac{\partial \phi}{\partial a_c} (\Omega\bar{B})_1 + (\bar{C}\Omega)_1 \frac{\partial \Psi}{\partial a_c} B_c C_2 H_d (\Omega\bar{B})_2$$

$$+ (\bar{C}\Omega)_1 \frac{\partial \Psi}{\partial a_c} B_c C_2 F_d] \Psi v_i u_i^* \} \qquad (3.45)$$

Substituting in Equations (3.10) and (3.11) for $\partial\phi/\partial a_c$ and $\partial\Gamma/\partial a_c$,

$$\frac{\partial \sigma_i}{\partial a_c} = \mathrm{Re} \cdot \mathrm{tr}\{\frac{1}{2} [(\bar{C}\Omega)_1 \Psi \frac{\partial A_c}{\partial a_c} (\phi + I)(\Omega\bar{B})_1 + (\bar{C}\Omega)_1 \Psi \frac{\partial A_c}{\partial a_c} \Psi B_c C_2 H_d (\Omega\bar{B})_2$$

$$+ (\bar{C}\Omega)_1 \Psi \frac{\partial A_c}{\partial a_c} \Psi B_c C_2 F_d] \Psi v_i u_i^* \}. \qquad (3.46)$$

Finally, we apply the rule in Equation (3.13) to obtain the matrix solution.

$$\frac{\partial \sigma_i}{\partial A_c} = \mathrm{Re}\{\frac{1}{2} [(\phi + I)(\Omega\bar{B})_1 \Psi v_i u_i^* (\bar{C}\Omega)_1 + \Psi B_c C_2 H_d (\Omega\bar{B})_2 \Psi v_i u_i^* (\bar{C}\Omega)_1$$

$$+ \Psi B_c C_2 F_d \Psi v_i u_i^* (\bar{C}\Omega)_1] \Psi\} \qquad (3.47)$$

Now, although the $\bar{A}$, $\bar{B}$, and $\bar{C}$ matrices are very different from those in Equations (3.36) and (3.37), their singular value gradients can be partitioned in exactly the same way. Doing this allows Equation (3.47) to be written in the very simple form

$$\frac{\partial \sigma}{\partial A_c^T} = \mathrm{Re}\{\frac{1}{2} [(\phi + I)(\frac{\partial \sigma_i}{\partial \bar{A}^T})_{11} + \bar{A}_{12}(\frac{\partial \sigma_i}{\partial \bar{A}^T})_{21} + \bar{B}_1(\frac{\partial \sigma_i}{\partial \bar{B}^T})_1] \Psi\} . \qquad (3.49)$$

It is interesting to note that these equations will also apply to the loop broken at the input case, although they must be applied to a completely different set of matrices! The second term in Equation

(3.49) will be zero for the input case, which causes it to reduce to the partitioned form of Equation (3.17) suggested at the end of Section 3.2.1.

Unfortunately, the B_c matrix gradients do not exhibit similar behavior; the output case requires a different equation to be derived. Since Γ appears in both the A and the B matrix, the following matrices must be plugged into Equation (2.39):

$$\frac{\partial \bar{A}}{\partial b_c} = \left[\begin{array}{c|c} 0 & \Psi \dfrac{\partial B_c}{\partial b_c} C_2 H_d \\ \hline 0 & 0 \end{array}\right] \tag{3.50}$$

$$\frac{\partial \bar{B}}{\partial b_c} = \left[\begin{array}{c} \Psi \dfrac{\partial B_c}{\partial b_c} C_2 F_d \\ \hline 0 \end{array}\right] \tag{3.51}$$

When this is done, and the matrices are reduced in a fashion analagous to that used in the derivation of $\partial\sigma_i/\partial a_c$, the following equation is obtained for $\partial\sigma_i/\partial B_c$:

$$\frac{\partial \sigma_i}{\partial B_c^T} = \left[C_2 F_d \left(\frac{\partial \sigma_i}{\partial \bar{B}^T}\right)_1 + C_2 H_d \left(\frac{\partial \sigma_i}{\partial \bar{A}^T}\right)_{21}\right] \Psi \ . \tag{3.52}$$

The assumptions of Section 3.2.1 apply here also, with the exception that in this case all feedbacks must go from the continuous to the digital system, instead of from the digital to the continuous system.

4. APPLICATION OF THE SVA TO A REPRESENTATIVE EXAMPLE

Sections 2 and 3 provide all the information necessary to do sensitivity analyses with respect to elements of the matrices $\bar{A}$, $\bar{B}$, and $\bar{C}$. In this section, the singular value analysis technique is illustrated using a simple example. This example will point out the necessity for a slight extension, to allow sensitivities to be performed with respect to parameters which do not appear explicitly in $\bar{A}$, $\bar{B}$, or $\bar{C}$.

4.1 AN INVERTED PENDULUM WITH A DIGITAL CONTROLLER

Consider the inverted pendulum example from Reference [15], pictured in Figure 4.1.

> The pivot of the pendulum is mounted on a carriage which can move in a horizontal direction. The carriage is driven by a small motor that at time t exerts a force $\mu(t)$ on the carriage. This force is the input variable of the system. Figure 4.2 indicates the forces and the displacements. The displacement of the pivot at time t is $d(t)$, while the angular rotation at time t of the pendulum is $\theta(t)$. The mass of the pendulum is m, the distance from the pivot to the center of gravity L, and the moment of inertia with respect to the center of gravity J. The carriage has mass M... F represents the friction coefficient [of the carriage]

See [15] for details on the derivation of the equations of motion. If the following variables are introduced:

$$L' = (J + mL^2)/mL; \tag{4.1}$$

$$\xi_1(t) = d(t) + L'\theta(t); \text{ and} \tag{4.2}$$

$$\xi_2(t) = \dot{\xi}_1(t); \tag{4.3}$$

then the following matrix equations approximate the system:

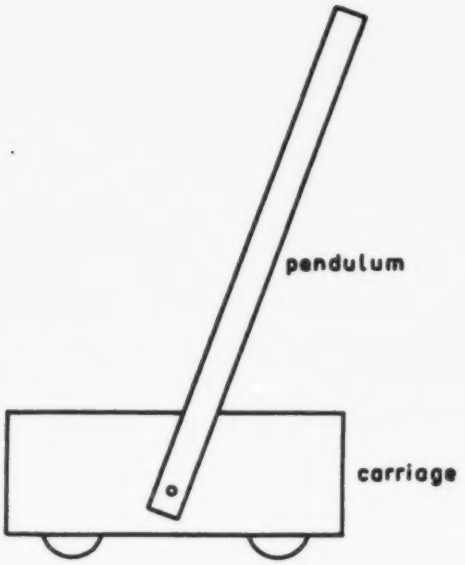


Figure 4.1: "An inverted pendulum positioning system" [15]

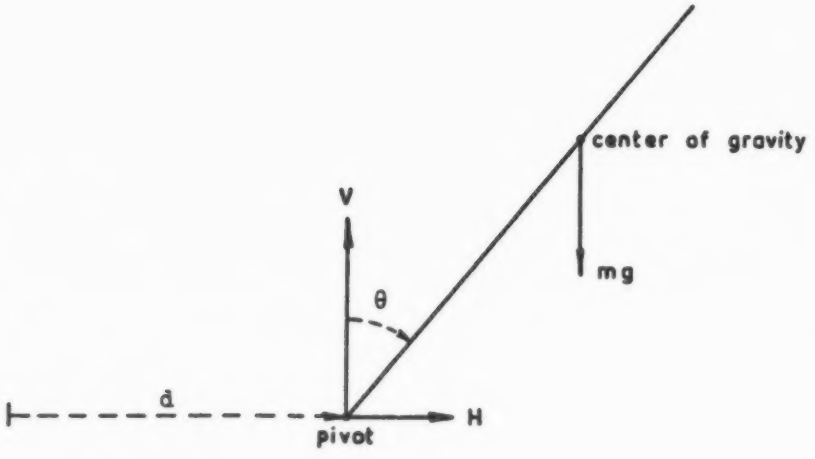


Figure 4.2: "Inverted pendulum: forces and displacements" [15]

$$\begin{bmatrix} \dot{d} \\ \ddot{d} \\ \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -F/M & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -g/L' & 0 & g/L' & 0 \end{bmatrix} \begin{bmatrix} \dot{d} \\ d \\ \xi_1 \\ \xi_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/M \\ 0 \\ 0 \end{bmatrix} \mu, \tag{4.4}$$

or $\dot{x} = Ax + B\mu$

To this system we add a continuous actuator whose dynamics are given in transfer function form as

$$\frac{BW}{s + BW}, \tag{4.5}$$

and we employ a proportional-plus-integral control law, implemented using a digital controller, to stabilize the system. See Figure 4.3. A digital lead-lag compensator will be used to enhance the system's stability and robustness.

This control system was chosen because it contains dynamics in both the discrete and the continuous domain. Also, the digital lead-lag provides a point of interest for sensitivity analyses because its pole- and zero- locations have not been optimized, so more robustness might be available if they are adjusted. Finally, this system is sufficiently complicated to illustrate many of the characteristics that are found in high-order linear models of real systems.

To study this control system, the $\bar{A}$, $\bar{B}$, and $\bar{C}$ matrices must first be found. This will be done explicitly here, because it is important to realize how control system parameters get combined into matrix elements. Computer programs to find $\bar{A}$, $\bar{B}$, and $\bar{C}$ given a block diagram are readily available.

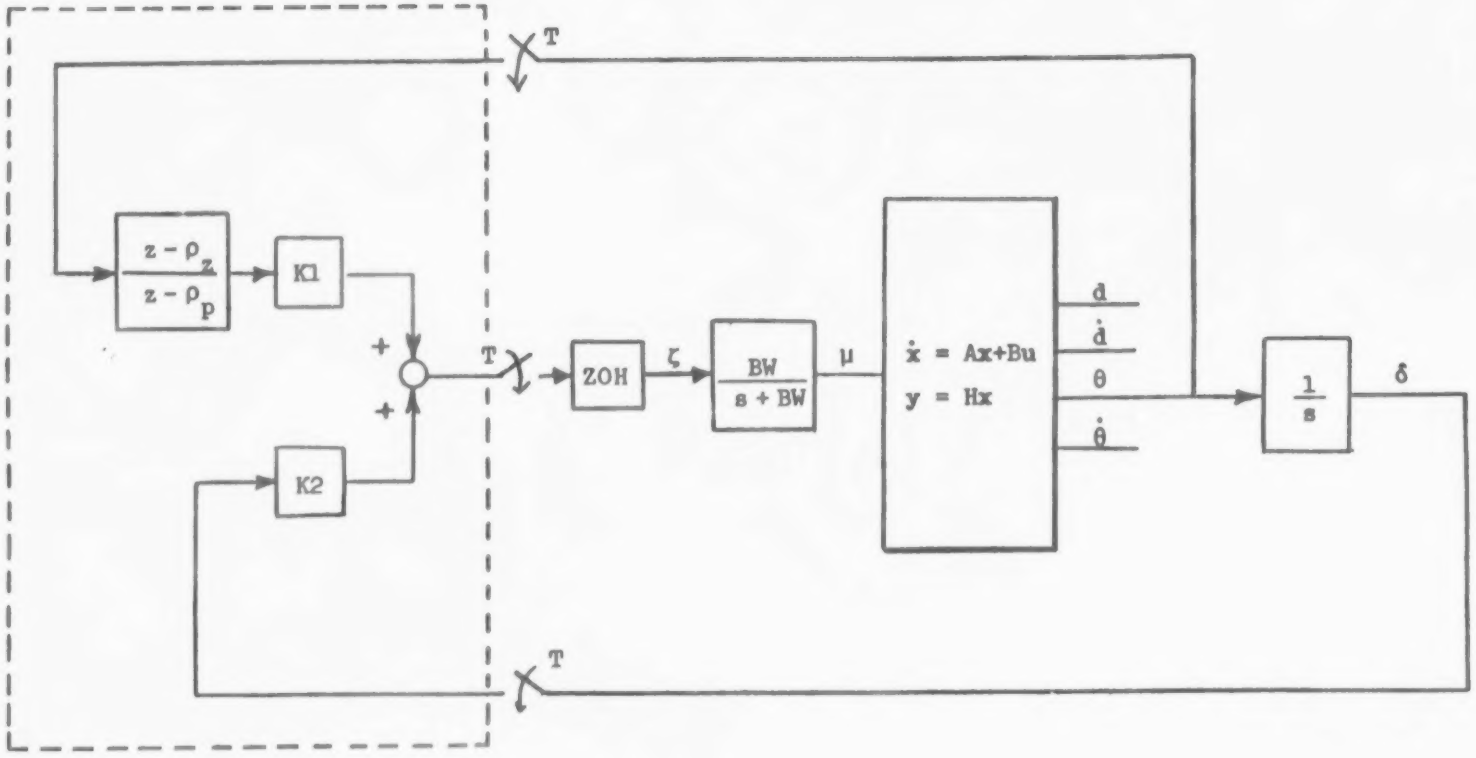


Figure 4.3: Proportional-plus-integral control system (with digital lead-lag for the inverted pendulum example of Kwakernaak and Sivan [15]

Since the control system in Figure 4.3 feeds back θ and its integral, this state must be reconstructed from the states in Equations (4.1)-(4.3).

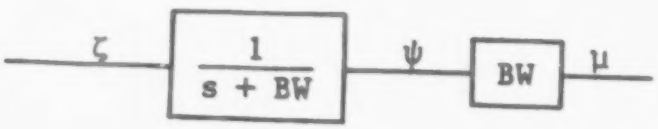
$$\xi_1 = d + L'\theta; \tag{4.3}$$

$$\therefore \quad \theta = 1/L'[\,-d + \xi_1]. \tag{4.6}$$

Equation (4.6) will be used in the development of the state matrix equations of the control system.

The first step in developing the state matrix equations is to add the continuous dynamics to the basic plant dynamics. The continuous dynamics in Figure 4.3 are the servo and the integrator. Each of these blocks will add an extra state to the state vector x.

The servo will be modeled as a first order filter coupled with a gain. (This method is used to conform to the methods of the program CONTROL, which is used extensively at Dryden Flight Research Facility, where these studies were carried out.) The new state variable will be called ψ, and the new control variable will be called ζ:



$$\psi = \frac{1}{s + BW}\,\zeta \text{ , so} \tag{4.7}$$

$$\psi s + BW\psi = \zeta \text{ .} \tag{4.8}$$

Taking the inverse Laplace transform of Equation (4.8) yields the equations necessary to augment ψ to the control system.

$$\dot{\psi} = -BW\psi + \zeta \; ; \tag{4.9}$$

$$\mu = BW\psi. \tag{4.10}$$

The integrator block is added in a similar manner, as follows, using Equation (4.6):

$$\delta = \frac{1}{s}\,\theta \; ; \tag{4.11}$$

$$\delta s = \theta \; ; \tag{4.12}$$

$$\dot{\delta} = \theta = \frac{1}{L'}\,[\xi_1 - d]. \tag{4.13}$$

The continuous dynamics are now completely modeled by the following equations:

$$\begin{bmatrix} \dot{x} \\ \dot{\psi} \\ \dot{\delta} \end{bmatrix} = \left[\begin{array}{cccc|c|c} & & A & & BW(B) & 0 \\ \hline & & 0 & & -BW & 0 \\ \hline -1/L' & 0 & 1/L' & 0 & 0 & 0 \end{array}\right] \begin{bmatrix} x \\ \psi \\ \delta \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \zeta \tag{4.14}$$

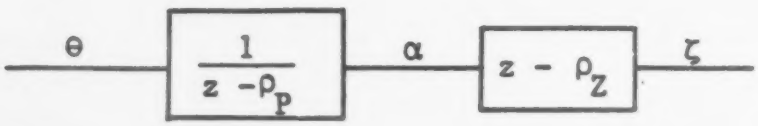
$$= A_c \begin{bmatrix} x \\ \psi \\ \delta \end{bmatrix} + B_c \zeta \; .$$

These matrices are next digitized, using Equations (3.2) and (3.3), so that the states are only known at the sample instants. Sample instants are spaced T seconds apart. The values of the states at each sample instant are indicated by the index k, and the state equations are reduced to the following matrix difference equations:

$$\begin{bmatrix} x_{k+1} \\ \psi_{k+1} \\ \delta_{k+1} \end{bmatrix} = \phi(A_c, T) \begin{bmatrix} x_k \\ \psi_k \\ \delta_k \end{bmatrix} + \Gamma(B_c, T)\zeta, \tag{4.15}$$

where the transition matrix $\phi(A_c,T)$ and discrete control-power matrix $\Gamma(B_c,T)$ are defined by Equations (3.2) and (3.3).

The state variables and matrices are now in the 'digital domain,' so the dynamics of the lead-lag filter and the feedback gain matrix can now be added to the system. The lead-lag filter reduces to difference equations using the methods of Reference 16. First, a new state variable, α, is defined. The input to the block is θ, and the output of the block is ζ, as shown in Figure 4.3.



$$\alpha = \frac{1}{z - \rho_p}\,\theta\ ; \tag{4.16}$$

$$\alpha(z - \rho_p) = \theta\ ; \tag{4.17}$$

$$\alpha z = \theta + \rho_p \alpha\ . \tag{4.18}$$

Taking the inverse z-transform of (4.18) yields

$$\alpha_{k+1} = \theta_k + \rho_p \alpha_k$$

$$= -\frac{1}{L'}\, d_k + \frac{1}{L'}\, \xi_{1_k} + \rho_p \alpha_k. \tag{4.19}$$

The feedback equation is

$$\zeta_k = K_1(\alpha z - \rho_z \alpha)\ , \tag{4.20}$$

which, taking the inverse Laplace transform and applying Equation (4.19), reduces to

$$\zeta_k = K_1(a_{k+1} - \rho_z a_k)$$

$$= K_1\left[-\frac{1}{L'} d_k + \frac{1}{L'} \xi_{1_k} + (\rho_p - \rho_z) a_k\right]. \quad (4.21)$$

The second feedback equation is simply (see Figure 4.3)

$$\zeta = K_2 \delta \quad (4.22)$$

or, using the inverse z-transform again to get the difference equation,

$$\zeta_k = K_2 \delta_k. \quad (4.23)$$

Combining Equations (4.15), (4.19), (4.22) and (4.23) yields the difference equations for the entire system.

$$\begin{bmatrix} d_{k+1} \\ \dot{d}_{k+1} \\ \xi_{1_{k+1}} \\ \xi_{2_{k+1}} \\ \psi_{k+1} \\ \delta_{k+1} \\ a_{k+1} \end{bmatrix} = \left[\begin{array}{ccccc|c|c} & & & & & 0 & \\ & & \phi & & & \dot{d}_k & \\ & & & & & 0 & \\ & & & & & 0 & \\ & & & & & 0 & \\ & & & & & 0 & \\ \hline -\frac{1}{L'} & 0 & \frac{1}{L'} & 0 & 0 & 0 & \rho_p \end{array}\right] \begin{bmatrix} d_k \\ \\ \xi_{1_k} \\ \xi_{2_k} \\ \psi_k \\ \delta_k \\ a_k \end{bmatrix} + \left[\begin{array}{c|c|c} & & \\ \Gamma & & \Gamma \\ & & \\ \hline 0 & & 0 \end{array}\right] \zeta_k$$

$$\zeta_k = \begin{bmatrix} -K_1 \frac{1}{L'} & 0 & K_1 \frac{1}{L'} & 0 & 0 & 0 & K_1(\rho_p - \rho_z) \\ 0 & 0 & 0 & 0 & 0 & K_2 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ \psi_k \\ \delta_k \\ a_k \end{bmatrix} \quad (4.24)$$

where $\phi = \exp$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{F}{M} & 0 & 0 & BW\frac{1}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -\frac{K}{L'} & 0 & -\frac{K}{L'} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -BW & 0 \\ -\frac{1}{L'} & 0 & \frac{1}{L'} & 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad \Gamma = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

This example has been executed in great detail to illustrate the point that often, even when the system is simple, the physical variables are not directly related to single matrix elements. For this reason, a slight extension to the singular value sensitivity analysis is required to get results which are useful. To illustrate, the following sensitivities will be performed:

$$\frac{\partial \underline{\sigma}}{\partial \rho_z}, \frac{\partial \underline{\sigma}}{\partial K_1}, \frac{\partial \underline{\sigma}}{\partial L'}, \text{ and } \frac{\partial \underline{\sigma}}{\partial BW}.$$

To compute these derivatives, one simply needs to compute the first-order Taylor series expansion using Equation (4.24).

$$\frac{\partial \underline{\sigma}}{\partial \rho_z} = -K_1 \frac{\partial \underline{\sigma}}{\partial c_{17}}; \tag{4.25}$$

$$\frac{\partial \underline{\sigma}}{\partial K_1} = -\frac{1}{L'} \frac{\partial \underline{\sigma}}{\partial c_{11}} + \frac{1}{L'} \frac{\partial \underline{\sigma}}{\partial c_{13}} - (\rho_p - \rho_k) \frac{\partial \underline{\sigma}}{\partial c_{17}}; \tag{4.26}$$

$$\frac{\partial \underline{\sigma}}{\partial L'} = \frac{1}{L'^2} \left[\frac{\partial \underline{\sigma}}{\partial a_{61}} + \frac{\partial \underline{\sigma}}{\partial a_{71}} - \frac{\partial \underline{\sigma}}{\partial a_{63}} - \frac{\partial \underline{\sigma}}{\partial a_{73}} + g \frac{\partial \underline{\sigma}}{\partial a_{41}} - g \frac{\partial \underline{\sigma}}{\partial a_{43}} + K_1 \frac{\partial \underline{\sigma}}{\partial c_{11}} - K_1 \frac{\partial \underline{\sigma}}{\partial c_{13}} \right]; \tag{4.27}$$

$$\frac{\partial \underline{\sigma}}{\partial BW} = -\frac{1}{M} \frac{\partial \underline{\sigma}}{\partial a_{25}} - \frac{\partial \underline{\sigma}}{\partial a_{55}}. \tag{4.28}$$

The first-order Taylor series expansions in Equations (4.25), (4.26), and (4.28) are exact. This is because the reduction of the block diagram in Figure 4.3 into state-space form required only linear combinations of block parameters and state-space elements.

No generalization of this result is attempted here, but experience indicates that if the denominator polynomials in the block diagram blocks have leading coefficients of one (or if the leading coefficients are never chosen as sensitivity parameters), then the vast majority of the elements of the A matrix are simple linear combinations of block diagram parameters. Thus if one chooses elements that actually appear as either matrix elements or block diagram polynomial coefficients, a first-order approximation will usually be exact. Equation (4.27) is inexact because L' appears as an inverse everywhere; it is <u>not</u> a state-space element per se. The approximation in Equation (4.27) will be accurate as long as L' is not near zero.

The general form for the equations presented above, when applied to any term in the block diagram describing the system (or any other parameter) is:

$$\frac{\partial \underline{\sigma}}{\partial p} = \Sigma \frac{\partial a_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial a_{ij}} + \Sigma \frac{\partial b_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial b_{ij}} + \Sigma \frac{\partial c_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial c_{ij}} . \qquad (4.29)$$

This equation has been implemented in the singular-value analysis program. The user must supply

$$\frac{\partial a_{ij}}{\partial p} , \frac{\partial b_{ij}}{\partial p} , \text{ and } \frac{\partial c_{ij}}{\partial p} .$$

for all appropriate i,j as inputs. For the studies in Section 5, these derivatives are computed using a modified version the program CONTROL, which computes the coefficients in Equation (4.29). Equation (4.29) has been found to be exact for all the parameters

which will be studied in section 5, because of the linearity (for many terms) of the transformation to state-space form. The terms

$$\frac{\partial \underline{\sigma}}{\partial a_{ij}}, \quad \frac{\partial \underline{\sigma}}{\partial b_{ij}}, \quad \text{and} \quad \frac{\partial \underline{\sigma}}{\partial c_{ij}}$$

in Equation (4.29) are simply elements of the matrix gradients

$$\frac{\partial \underline{\sigma}}{\partial \underline{A}^T}, \quad \frac{\partial \underline{\sigma}}{\partial \underline{B}^T}, \quad \text{and} \quad \frac{\partial \underline{\sigma}}{\partial \underline{C}^T}.$$

The equations for these gradients were derived in Sections 2 and 3.

4.2 STEP-BY-STEP APPLICATION OF THE SVA

Equations from subsection 4.1 as well as Sections 2 and 3 must be combined to do the singular value analysis. Once the gradient plots are computed, some method for organizing and presenting the results is necessary, because of the large number of plots generated. In this subsection, the procedures used by the program SVA are delineated step-by-step to bring together all the necessary components from the previous sections. The inverted pendulum example will be used to help clarify the details of the analysis.

4.2.1 Setting up the Matrices

A complete robustness analysis of a control system requires that the singular values and their gradients be computed at both the plant input and the plant output (or, equivalently, at the plant input and at the controller input). This simply means that the

analysis must be done for two sets of the matrices $\bar{A}$, $\bar{B}$, $\bar{C}$. In this section we will present how these matrices are constructed. In Sections 4.2.2 and 4.2.3, all the steps are the same whether the system is being analyzed for robustness to uncertainty at the plant input or at the plant output (controller input). Thus no distinction is made between the two cases in these sections; any $\bar{A}$, $\bar{B}$, and $\bar{C}$ will do. The only exception is that gradients with respect to $\bar{B}$ matrix elements are different for the input and output cases when the system is digital, as discussed in Section 3.2; this exception will be noted below. Section 3.2 presents a detailed explanation of how the $\bar{A}$, $\bar{B}$, and $\bar{C}$ matrices must be formed when the control system is digital. For the continuous case, if the system is described by the equations

PLANT:
$$\begin{aligned} \dot{x}_p &= A_p x_p + B_p u_p \\ y_p &= H_p x_p + F_p u_p \end{aligned} \tag{4.30}$$

AUGMENTATION:
$$\begin{aligned} \dot{x}_a &= A_a x_a + B_a u_a \\ y_a &= H_a x_a + F_a u_a \end{aligned} \tag{4.31}$$

INTERCONNECTIONS:
$$\begin{aligned} u_a &= C_1 y_p \\ u_p &= C_2 y_a, \end{aligned} \tag{4.32}$$

and if all dimensions are compatible, the necessary matrices for the input case are

$$\bar{A} = \left[\begin{array}{c|c} A_p & 0 \\ \hline B_a C_1 H & A_a \end{array}\right], \qquad \bar{B} = \left[\begin{array}{c} B_p \\ \hline B_a C_1 F_p \end{array}\right],$$
$$\bar{C} = -(I - C_2 F_a C_1 F_p)^{-1}[\, C_2 F_a C_1 H_p \mid C_2 H_a \,]; \tag{4.33}$$

and the necessary matrices for the output case are

$$\bar{A} = \left[\begin{array}{c|c} A_p & B_pC_2H_a \\ \hline 0 & A_a \end{array}\right], \qquad \bar{B} = \left[\begin{array}{c} B_pC_2F_a \\ \hline B_a \end{array}\right],$$

$$\bar{C} = -(I - C_1F_pC_2F_a)^{-1}[\ C_1H_p \mid C_1F_pC_2H_a\ ]\ . \tag{4.34}$$

Note that the minus signs in front of the $\bar{C}$ matrices arise from the fact that the analysis assumes negative feedback.

To "scale" the system, one simply needs to introduce a scaled input vector u_s,

$$u_s = Du.$$

where u is the control input vector for the total system: u_p in the input case and u_a in the output case. D, the scaling matrix, should be square and invertable (it is also desirable for D to be real and diagonal, to preserve the intuitive idea of changing the "units" being used, although References [17] and [20] suggest 'optimum' scalings which do not preserve this notion). D can be introduced into the control system by letting $u=D^{-1}u_s$ as follows:

$$\dot{x} = \bar{A}x + \bar{B}D^{-1}u_s \ ;$$

$$D^{-1}u_s = -\bar{C}x, \quad \text{or}$$

$$u_s = -D\bar{C}x.$$

Thus the analysis can proceed as usual, with the scaled system represented by $\bar{A}_s = \bar{A}$, $\bar{B}_s = \bar{B}D^{-1}$, and $\bar{C}_s = D\bar{C}$. It is easily verified and intuitively obvious that the closed-loop properties of the control system are unaltered by this type of system scaling. However, except in the case of diagonal perturbations, the singular-value robustness results from a scaled system are valid only for that system [17]. On the other hand, along the diagonal of the

perturbation matrix L, since scaling can not effect the allowable multiplication factors, singular value results obtained from the scaled system are valid for the unscaled system (see Section 2.3 for a complete discussion of scaling).

4.2.2 Computations at Each Frequency

To get a plot in which ω is the independent variable, the gradient matrices and partial-derivative expansions must be computed at each ω. This is a repetitive process; the same steps are followed for each value of ω. We present the computation for one value of ω, for a digital system.

A> Let $z = e^{j\omega T} = \cos(\omega T) + j \cdot \sin(\omega T)$.

B> Compute Ω, where

$$\Omega = (Iz - \bar{A})^{-1}. \tag{4.35}$$

C> Compute the return difference matrix for the control system,

$$RDM = (\bar{C}\Omega\bar{B} + I). \tag{4.36}$$

D> Perform the singular value decomposition on the return difference matrix. The resulting matrices are S, a diagonal matrix of singular values, ordered from largest to smallest; and U and V, the matrices containing the right and left singular vectors. The columns u_i and v_i of U and V respectively correspond to the $\sigma_i = s_{ii}$ in the matrix S by the following equations from Section 2:

$$(I + HG)v_i = u_i\sigma_i \quad ; \tag{2.31}$$

$$(I + HG)^* u_i = v_i\sigma_i \ . \tag{2.32}$$

E> Extract σ_n, u_n and v_n, where n = the dimension of the square matrix A. For the pendulum example, n = 7. Since the σ_i are ordered from largest to smallest, $\sigma_n = \underline{\sigma}$.

F> Compute the gradients of the digital-domain matrices using the equations from Section 2.

$$\frac{\partial \underline{\sigma}}{\partial \bar{A}^T} = \mathrm{Re}[\Omega \bar{B} \Psi_n u_n^* \bar{C} \Omega] \; ; \tag{2.46}$$

$$\frac{\partial \underline{\sigma}}{\partial \bar{B}^T} = \mathrm{Re}[\Psi_n u_n^* \bar{C} \Omega] \; ; \tag{2.48}$$

$$\frac{\partial \underline{\sigma}}{\partial \bar{C}^T} = \mathrm{Re}[\Omega \bar{B} \Psi_n u_n^*] \; . \tag{2.50}$$

G> Extract the necessary quadrants of $\partial \underline{\sigma}/\partial \bar{A}^T$ and $\partial \underline{\sigma}/\partial \bar{B}^T$, and use them in the equations for $\partial \underline{\sigma}/\partial A_c^T$ and $\partial \underline{\sigma}/\partial B_c^T$ which were derived in Section 3 (A_c and B_c for the pendulum example appear in Equation 4.14). For $\partial \underline{\sigma}/\partial B_c^T$, the necessary equation depends on whether the L matrix is at the plant or controller input.

$$\frac{\partial \underline{\sigma}}{\partial A_c^T} = \mathrm{Re}\left\{\frac{1}{2}\left[(\phi + I)\left(\frac{\partial \underline{\sigma}}{\partial \bar{A}^T}\right)_{11} + \bar{A}_{12}\left(\frac{\partial \underline{\sigma}}{\partial \bar{A}^T}\right)_{21} + \bar{B}_1\left(\frac{\partial \underline{\sigma}}{\partial \bar{B}^T}\right)_1\right]\Psi\right\} \; . \tag{3.49}$$

for L at the plant input,

$$\frac{\partial \underline{\sigma}}{\partial B_c^T} = \left(\frac{\partial \underline{\sigma}}{\partial \bar{B}^T}\right)_1 \Psi^T \; , \tag{3.22}$$

and for L at the plant output,

$$\frac{\partial \underline{\sigma}}{\partial B_c^T} = \left[C_2 F_d \left(\frac{\partial \underline{\sigma}}{\partial \bar{B}^T}\right)_1 + C_2 H_d \left(\frac{\partial \underline{\sigma}}{\partial \bar{A}^T}\right)_{21}\right]\Psi \; . \tag{3.52}$$

The gradients $\partial \underline{\sigma}/\partial \bar{C}^T$ are valid without changes.

H> The matrices in Equations (3.49), (3.22), and (3.52) are of the same dimensions as the quadrants which were extracted in from $\partial \underline{\sigma}/\partial \bar{A}^T$ and $\partial \underline{\sigma}/\bar{B}^T$ in step G. Insert the results of step G into the matrices computed in step F. The resulting matrices contain all the physically significant sensitivities, and are illustrated for the pendulum example in Figure 4.4.

$$\left(\frac{\partial \underline{\sigma}}{\partial \bar{A}^T}\right)^T = \begin{bmatrix} X & X & X & X & X & X \\ X & \frac{\partial \underline{\sigma}}{\partial(-\frac{F}{M})} & X & X & \frac{\partial \underline{\sigma}}{\partial(-\frac{BW}{M})} & X \\ X & X & X & X & X & X \\ \frac{\partial \underline{\sigma}}{\partial(-\frac{g}{L'})} & X & \frac{\partial \underline{\sigma}}{\partial(\frac{g}{L'})} & X & X & X \\ X & X & X & X & \frac{\partial \underline{\sigma}}{\partial(-BW)} & X \\ \frac{\partial \underline{\sigma}}{\partial(-\frac{I}{L'})} & X & \frac{\partial \underline{\sigma}}{\partial(\frac{I}{L'})} & X & X & X \\ \frac{\partial \underline{\sigma}}{\partial(-\frac{I}{L'})} & X & \frac{\partial \underline{\sigma}}{\partial(\frac{I}{L'})} & X & X & X \end{bmatrix}$$

$$\left(\frac{\partial \underline{\sigma}}{\partial \bar{C}^T}\right)^T = \begin{bmatrix} \frac{\partial \underline{\sigma}}{\partial(-\frac{K_1}{L'})} & X & \frac{\partial \underline{\sigma}}{\partial(\frac{K_1}{L'})} & X & X & X \\ X & X & X & X & X & X \end{bmatrix}$$

Figure 4.4: Results of the singular value anaysis when applied to the pendulum example
(X = this element contains a gradient with no physical significance.)

I> Construct the desired gradients using Equation (4.29)

$$\frac{\partial \underline{\sigma}}{\partial p} = \Sigma \frac{\partial a_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial a_{ij}} + \Sigma \frac{\partial b_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial b_{ij}} + \Sigma \frac{\partial c_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial c_{ij}} \qquad (4.29)$$

The results of going through steps A> through I> at each ω for the pendulum example will be presented in Section 4.3.

4.2.3 Data Reduction of the $\underline{\sigma}$-Gradient Plots

If we repeat the steps in Section 4.2.1 for a range of ω's, we will end up with $\underline{\sigma}$-gradient plots for all the parameters of interest. The next step is to construct a table of the parameters in their order of 'importance,' and to display enough information so that the table can be interpreted. To construct this table, we must search the frequency range to find the frequency at which each variable exerts its strongest influence. Again we present a step-by-step procedure.

A> For each parameter, perform the following steps:

(1) At each frequency, compute the $\Delta p/p$ (or percent variation in p) required to drive $\underline{\sigma}(\omega)$ to some minimum allowable $\underline{\sigma}$ ($\underline{\sigma}_{MA}$)

$$\frac{\Delta p}{p} = \left[\frac{(\underline{\sigma} - \underline{\sigma}_{MA})}{\left(\frac{\partial \underline{\sigma}}{\partial p} \cdot p\right)} \right] \qquad (4.37)$$

(2) Find the minimum $\Delta p/p$ for all ω.

Steps (1) and (2) are simply the application of Equation (2.51)

$$\min_{0<\omega<=} \left[\frac{\Delta p}{p}\right] = \min_{0<\omega<=} \left[\frac{\underline{\sigma} - \underline{\sigma}_{MA}}{\frac{\partial \sigma}{(\partial p/p)}}\right]. \qquad (2.51)$$

B> Compare the $\Delta p/p$'s for each parameter to set up the table.

The resulting table contains the following information:

1) The name of the parameter

2) The frequency at which it exerts its strongest effect

3) $\underline{\sigma}$ at this frequency

4) $\partial\underline{\sigma}/(\partial p/p)$, the 'normalized singular-value gradient,' at the freqency in column 2. The normalization is performed so that different parameters can be compared to one another.

5) The $\Delta p/p$, in percent, required to drive the $\underline{\sigma}$-plot to $\underline{\sigma}_{MA}$ at the frequency in column 2, as shown in Figure 2.9.

6) The percent variation in $\underline{\sigma}$ needed (at the frequency shown in column 2) to drive it to $\underline{\sigma}_{MA}$. This column helps to identify when the linearity assumption might be invalid.

The table described above is very useful when many parameters must be compared; however the $\underline{\sigma}$-gradient plots still contain vital information and should not be overlooked, as we will see in the next section.

F-2

4.3 APPLICATION OF THE SVA TO THE PENDULUM EXAMPLE

The results of conducting the analysis described in section 4.2 are presented in this section for the pendulum example, given the plant and controller characteristics given below.

plant:		controller:	
F/M	$= 1\ s^{-1}$	K1	= 70
1/M	$= 1\ kg^{-1}$	K2	= 13
L'	= .842 m	ρ_z	= .8
g/L'	$= 11.65\ s^{-2}$	ρ_p	= .1
BW	= 10 rad/s		

Section 4.3.1 discusses the $\underline{\sigma}$-plots for this system, and the effects of scaling on it. Section 4.3.2 gives examples of $\underline{\sigma}$-gradients.

4.3.1 $\underline{\sigma}$-Plots for the Pendulum Example

Figures 4.5 and 4.6 are the $\underline{\sigma}$-plots for the unscaled system. Figure 4.5 measures robustness to uncertanties at the plant input, while Figure 4.6 presents robustness at the plant output. The fact that these plots are very similar is in no way typical; it probably stems both from the simplicity of the control system, and from the fact that there are only two plant outputs and one plant input. The low order of the system also accounts for the fact that there is very little difference between the singular values and the eigenvalues.

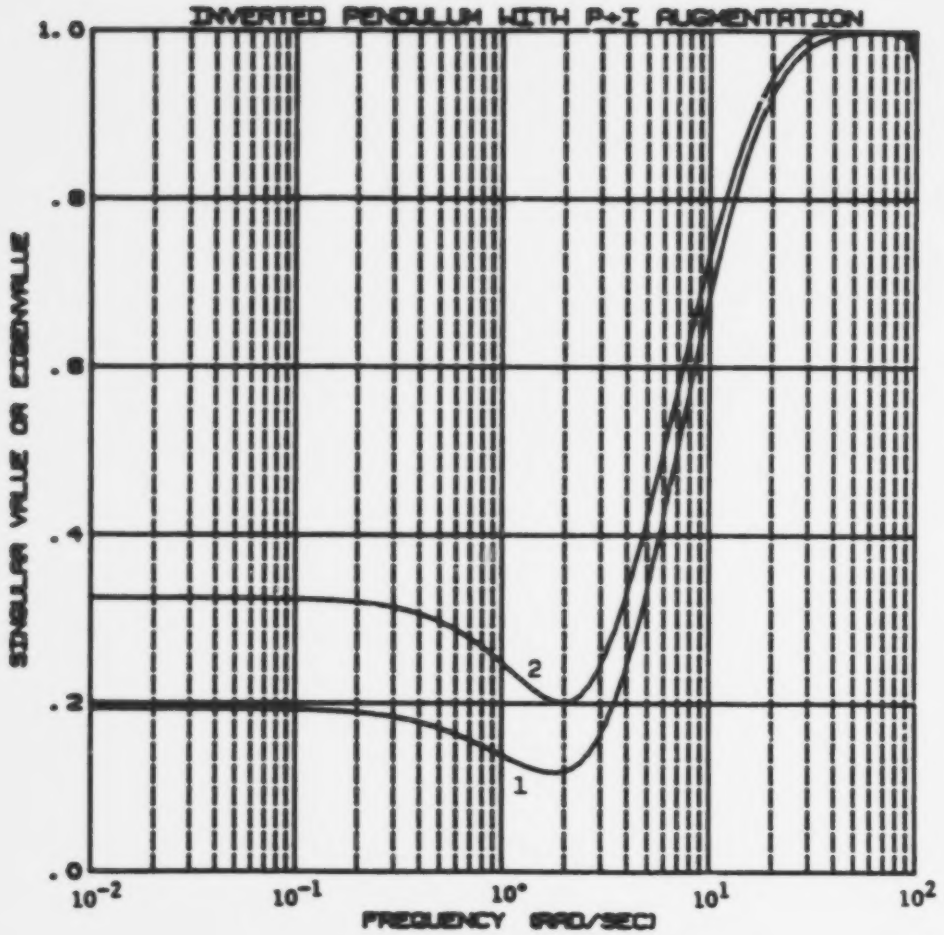


Figure 4.5: Plot of 1) minimum singular value and 2) minimum eigenvalue for the pendulum example (unperturbed control system) ; input node

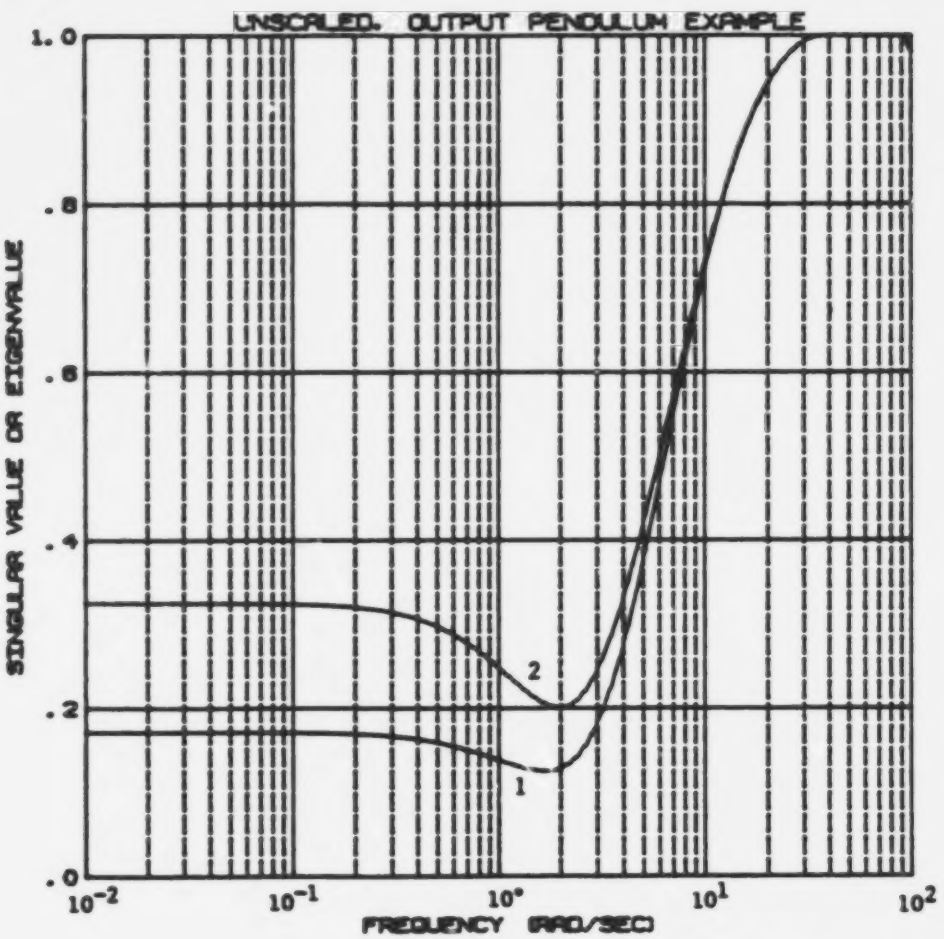


Figure 4.6: Plot of 1) minimum singular value and 2) minimum eigenvalue for the pendulum example (unperturbed control system); output node

The robustness of this system would be considered unacceptable for most applications, even if scaling could be used to drive the minimum singular value up to the level of the minimum eigenvalue. Nevertheless, we will use this case to illustrate the technique and effect of scaling.

At the plant input, the scaling should be chosen so as to maximize $\underline{\sigma}$ across the frequency range. To this end, we plot $\underline{\sigma}$ for D matrices of the form

$$\begin{bmatrix} d & 0 \\ 0 & 1 \end{bmatrix} ,$$

where d takes on various values. This scheme is general enough to allow all real, diagonal scalings to be studied. Figure 4.7 shows the results for d= .25, .5, .714, 1.25 and 2.5. It can be seen that in this example, very little improvement in σ_{min} is available from this type of scaling. In some cases, however, drastic improvements can be achieved in this way [17].

At the output, the scaling should be chosen so as to normalize the output magnitude. For instance, by looking at the time histories of our baseline pendulum example control system design, we can choose a scaling D which will cause the scaled outputs, θ_s and δ_s, to vary between zero and one for a 5 degree initial condition on θ. This scaling is

$$\begin{bmatrix} 5 & 0 \\ 0 & 1.595 \end{bmatrix}^{-1} ,$$

which reflects the maximum magnitudes of θ and δ in a time history.

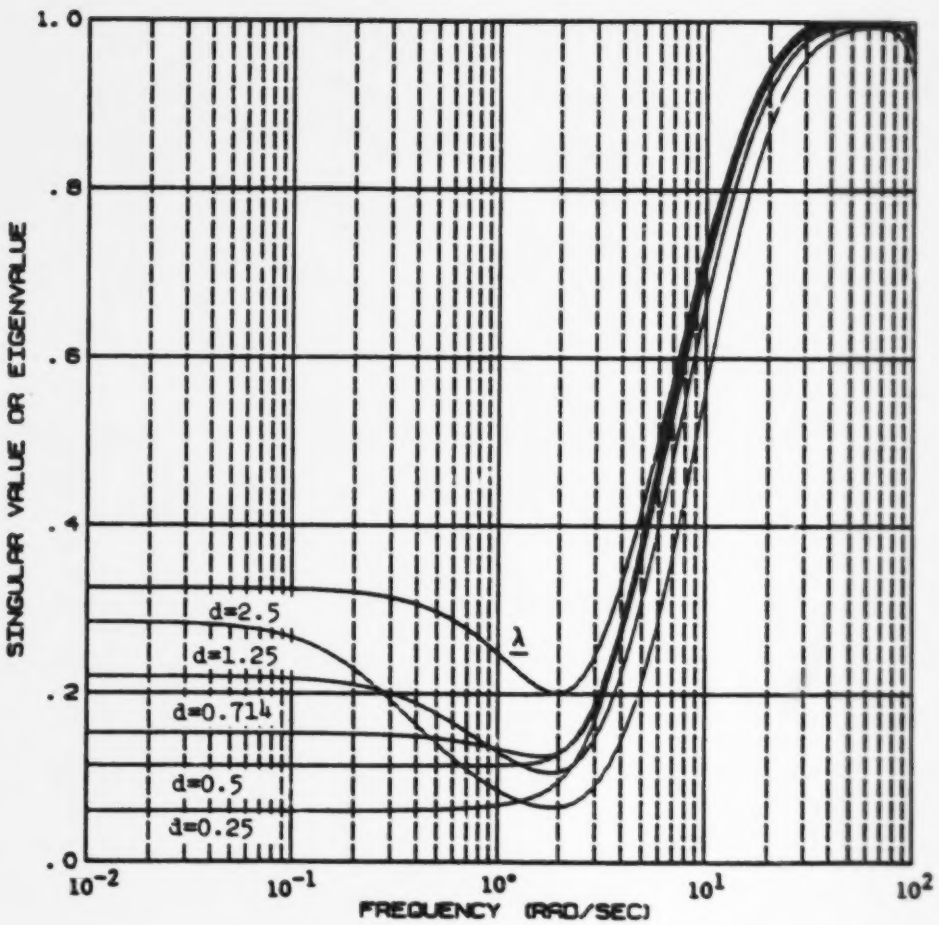


Figure 4.7: $\bar{\sigma}$-plots at the input node of the pendulum example for various scalings. $\underline{\lambda}$-plot is included to show that it is an upper bound.

More complex systems would be more difficult to scale, because relative output magnitudes depend on the maneuver being performed. The main purpose in this type of scaling is not to account for all possible maneuvers, but to normalize out the effects of large differences in units. In this case, as seen in Figure 4.8, the system actually looks more sensitive when the system is scaled (compare Figure 4.6). Since this plot indicates sensitivity at the output, Figure 4.8 might indicate the need for very accurate measurement devices, with very little noise, for this control system. The fact that an integrator would probably be used to implement the measurement of δ (see Figure 4.3) would probably solve this problem, since the scaled system puts more emphasis on δ than does the unscaled system.

4.3.2 $\underline{g}$-Gradient Plots for the Pendulum Example

The $\underline{g}$-gradient tables (at the input and at the output) for the unscaled pendulum example are given in Tables 4.1 and 4.2. They result from applying the steps of Section 4.2. $\underline{g}$-gradient plots for the parameters L', K1, ρ_z, and BW are shown in Figures 4.9 and 4.10, for the input case.

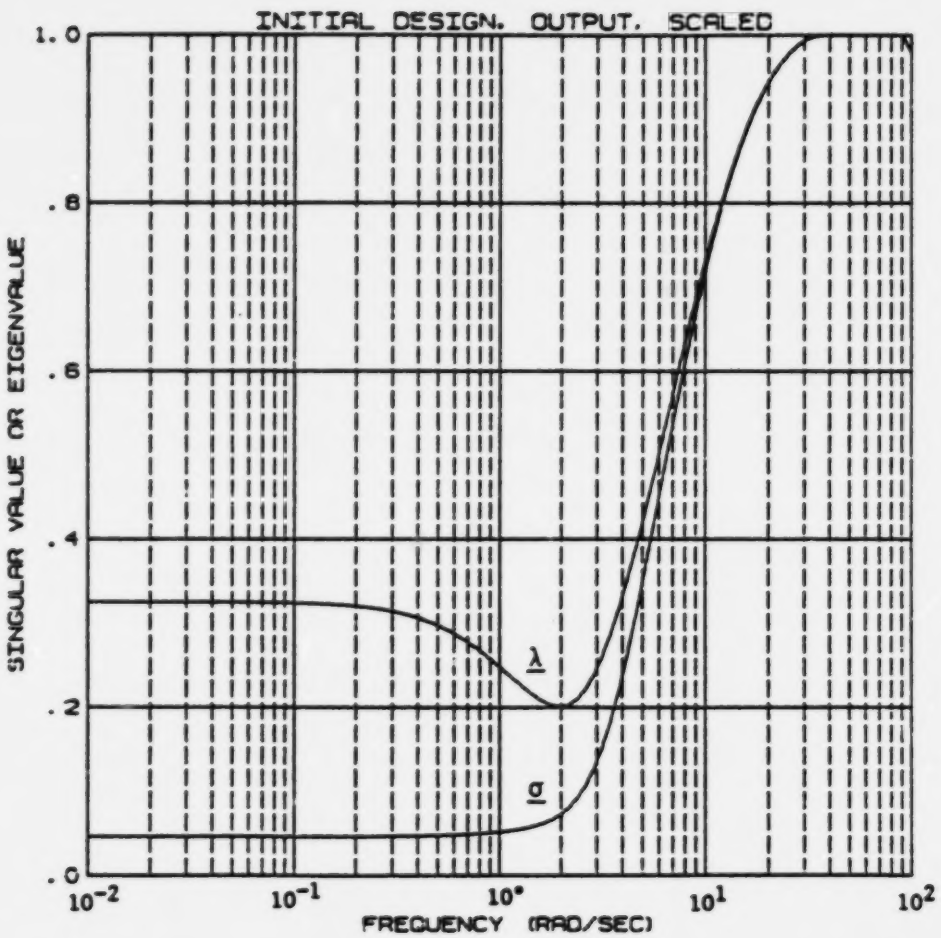


Figure 4.8: $\underline{\sigma}$-plot/$\underline{\lambda}$-plot at the output node for the pendulum example with scaling

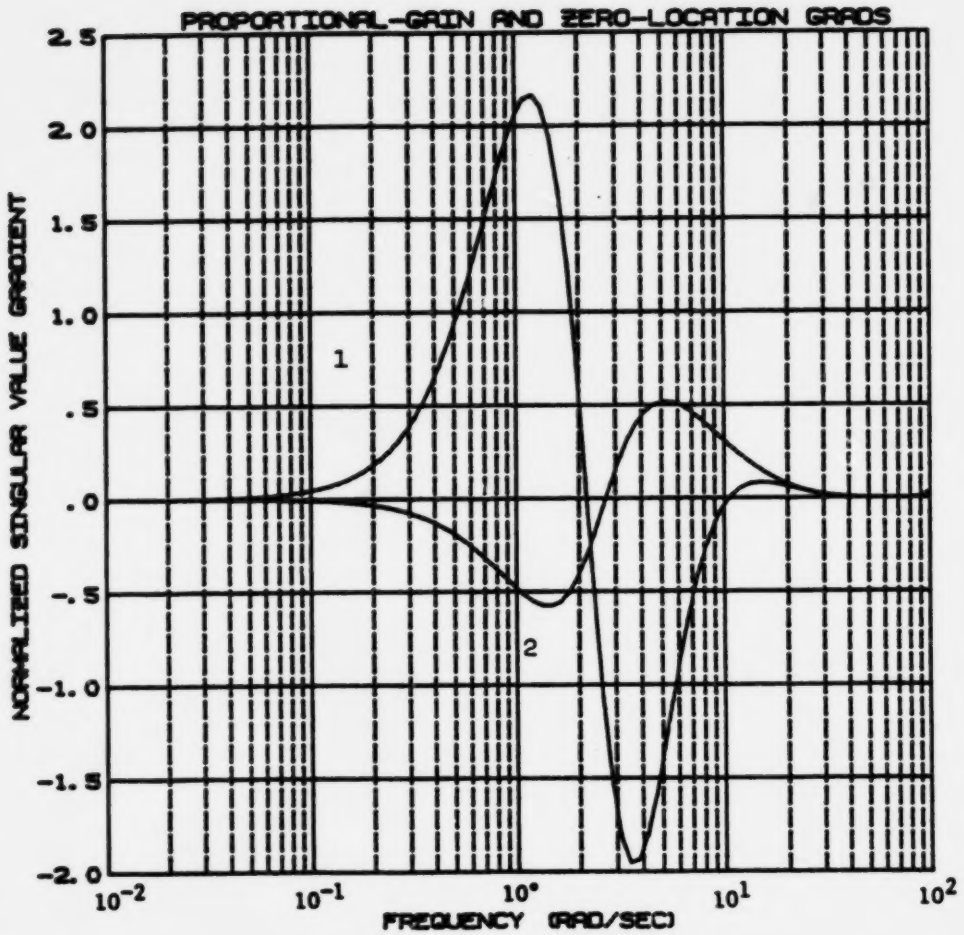


Figure 4.9: Plot of the singular value gradients with respect to 1) proportional gain (K1) and 2) compensator zero location ($-\rho_z$) for the pendulum example (unperturbed control system) input node

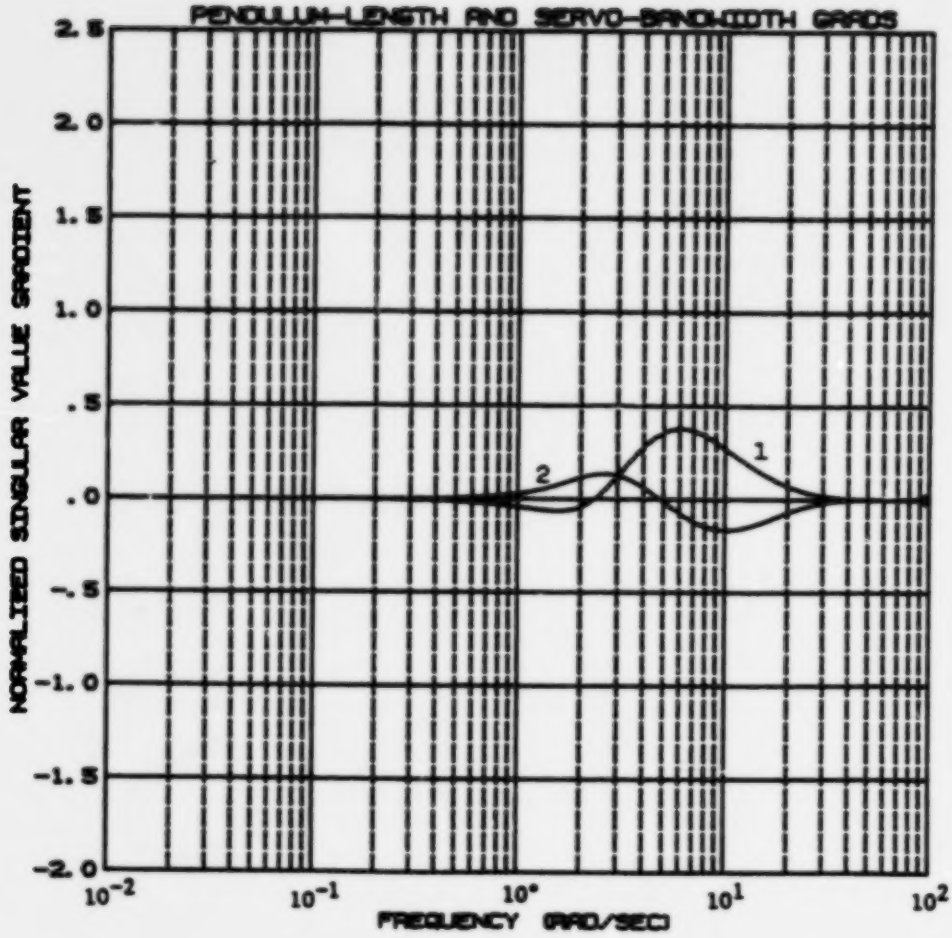


Figure 4.10: Plot of the singular value gradients with respect to 1) pendulum characteristic length (L') and 2) servo bandwidth (BW) for the pendulum example (unperturbed control system) input node

Table 4.1: Parameter sensitivities for the pendulum example at the input node: variations necessary to drive $\underline{\sigma}_{min}$ to .06 (half the original value)

param.(p)	ω_{min}	$\underline{\sigma}(\omega_{min})$	$\partial\underline{\sigma}/(\partial p/p)$	Δp	$\Delta\underline{\sigma}$
ρ_z	1.33	.12	2.10	-3%	-52%
KI	1.62	.12	-.56	11%	-49%
BW	2.15	.12	.13	-49%	-51%
L'	3.83	.23	.26	-66%	-74%

Table 4.2: Parameter sensitivities for the pendulum example at the output node: variations necessary to drive $\underline{\sigma}_{min}$ to .06 (half the original value)

param.(p)	ω_{min}	$\underline{\sigma}(\omega_{min})$	$\partial\underline{\sigma}/(\partial p/p)$	Δp	$\Delta\underline{\sigma}$
ρ_z	1.33	.13	2.25	-3%	-53%
KI	1.62	.13	-.62	11%	-52%
BW	2.15	.13	.14	-53%	-55%
L'	3.83	.26	.27	-73%	-77%

Definitions:

param: Parameter for which gradient has been taken

ω_{min} : Frequency at which this parameter has maximum effect

$\underline{\sigma}(\omega_{min})$: Singular value at ω_{min}

$\partial\underline{\sigma}/(\partial p/p)$: Singular-value gradient at ω_{min}

Wp: Percentage change in p necessary to drive $\underline{\sigma}_{min}$ to .06

$\Delta\underline{\sigma}$: Percentage change in $\underline{\sigma}$ necessary at ω_{min} to get to .06

5. APPLICATION OF THE SVA TO A REAL SYSTEM

This section is devoted to applying the results of the previous sections to the linear model of a real system. Section 5.1 compares $\underline{\sigma}$-plots with frequency response results. Section 5.2 presents some $\underline{\sigma}$-gradient results and compares them to other sensitivity results. In this section, the issues of scaling and of output singular values and their gradients are not addressed. Scaling is not necessary in the cases presented here because the singular values are very close to the eigenvalues in the critical frequency ranges. Output singular values are not presented here because the intent is not to do a complete analysis of the control system, but to demonstrate the feasibility of applying the SVA to real problems.

5.1 $\underline{\sigma}$-PLOTS APPLIED TO A REAL SYSTEM

References [1] and [3] discuss the differences between classical PM and GM and the PM and GM obtained from $\underline{\sigma}$-plots. Both references compared SISO (classical) and MIMO ($\underline{\sigma}$-plot) stability margins for low-order systems. In this section, similar comparisons will be done for two high-order control systems, which together make up the linear model of the primary flight control system of the X-29 Advanced Technology Demonstrator.

The two systems we will use are the lateral-directional and the longitudinal control systems for the X-29. They both represent a high degree of augmentation; and, in the primary mode, they are both

digital. The linear model for the longitudinal aircraft dynamics, with actuators, sensors, and control laws, is 48th order. The lateral-directional model is a 35th order system. Appendix B contains geometric data and dynamic matrices for the X-29.

5.1.1 X-29 Longitudinal Mode

Figure 5.1 shows a simplified block diagram for the X-29 longitudinal mode. The $\underline{\sigma}$-plot will be computed for the system with the control loops "broken" at each of two locations. At point A in Figure 5.1 the system is SISO. At point B it is MIMO. By analyzing the $\underline{\sigma}$-plots of the system at both of these points, and comparing the results to a classical Bode analysis, some idea can be formed about the additional conservativeness introduced when a system is analyzed in a MIMO sense. "Breaking" the loop at a different point primarily effects how the $\bar{C}$ matrix, the feedback gain matrix, will be defined. The Bode analysis can of course only be applied to the SISO system.

It is interesting to note that in this special case (i.e., the system branches from a single channel into multiple channels), the eigenvalue-plot (or $\underline{\lambda}$-plot) of the MIMO system should match the $\underline{\sigma}$-plot of the SISO system exactly. This is because $\underline{\lambda}$-plots measure relative stability when the disturbances are uniform; i.e., all loops change in gain and phase simultaneously. This is exactly the effect that breaking the loop at the SISO point has: the disturbances at the three separate loops will be equal.

Figure 5.2 is the $\underline{\sigma}$-plot for the SISO system, for the X-29 at 15,000 feet and a Mach number of .9. The $\underline{\sigma}$-plot of this system is

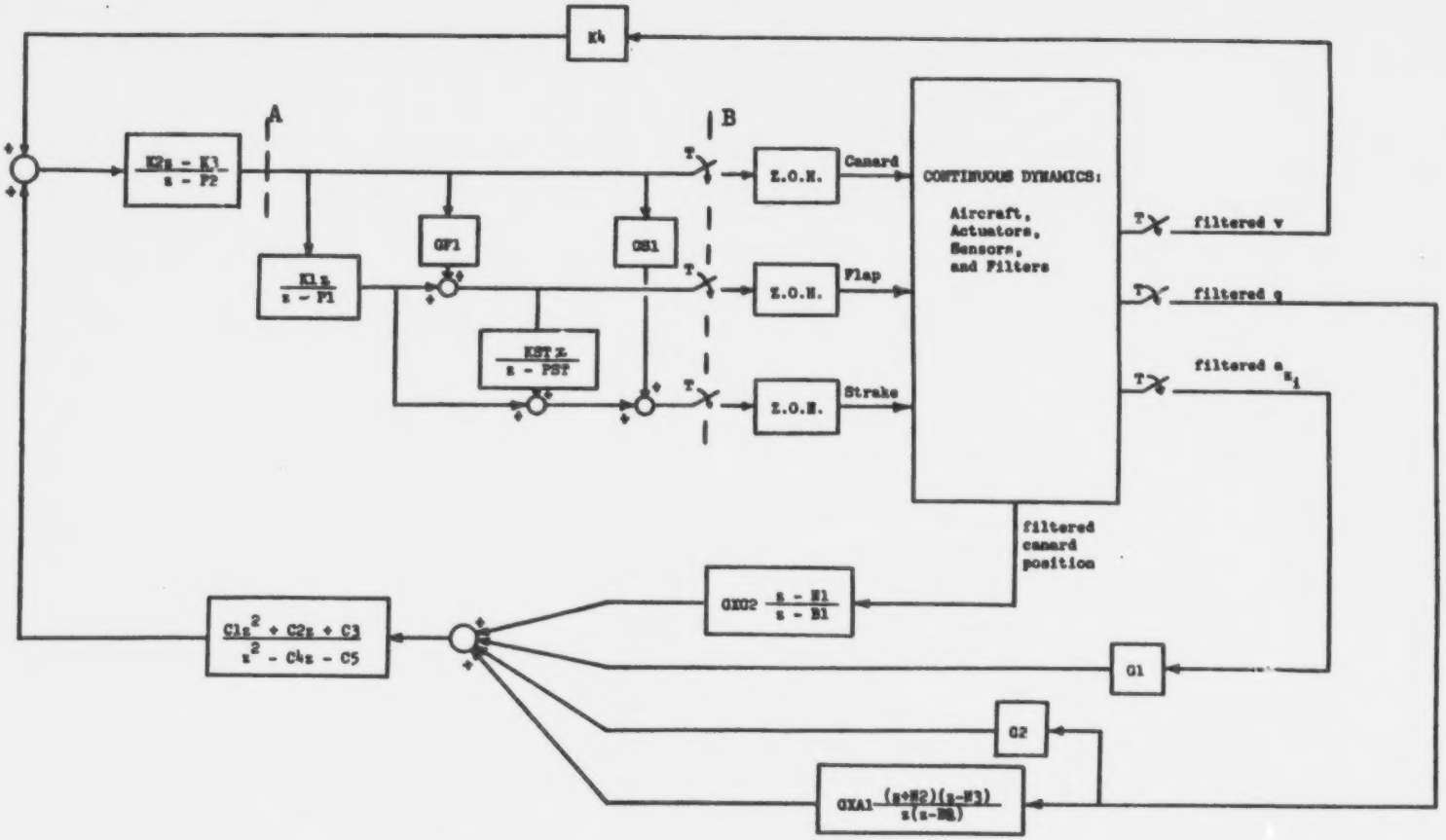


Figure 5.1: Simpified block diagram of the X-29 longitudinal primary flight control system

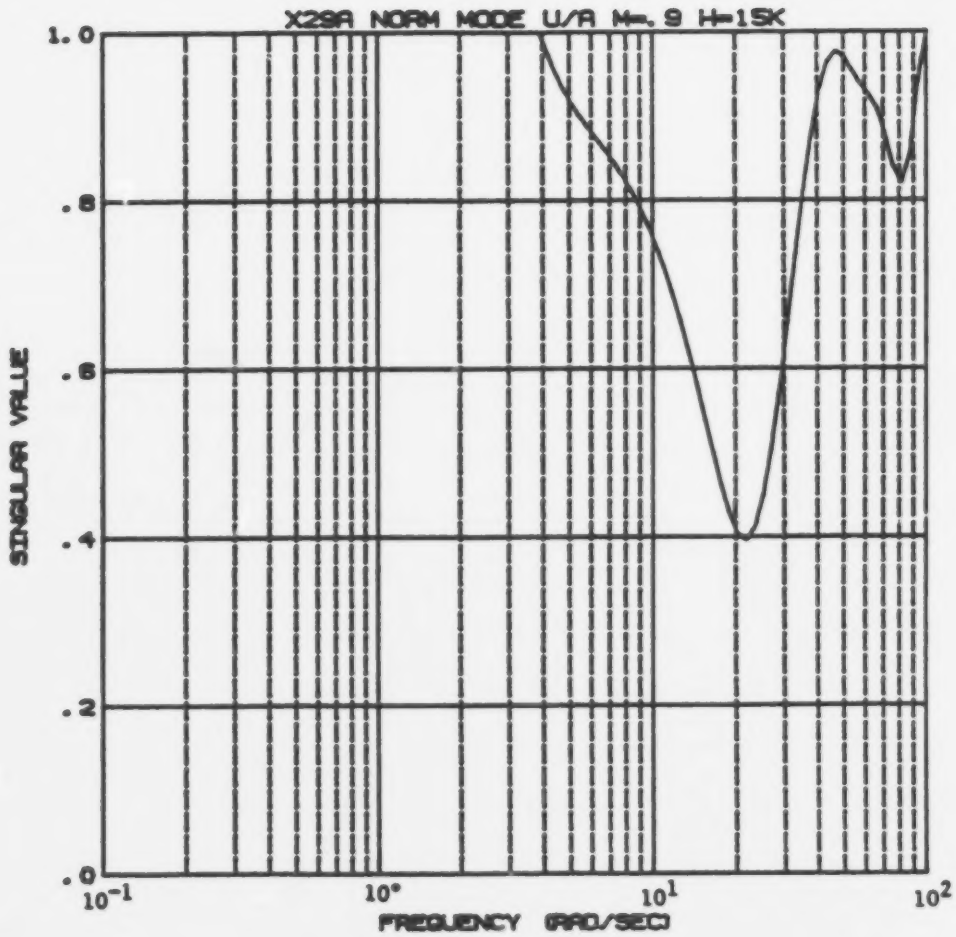


Figure 5.2: Singular value plot for the X-29 longitudinal mode at M=.9 and H=15,000 ft., with the loop broken at point A in Figure 5.1

the same as the $\underline{\lambda}$-plot, because SISO disturbances are necessarily uniform. Figure 5.3 is the $\underline{\sigma}$-plot/$\underline{\lambda}$-plot for the MIMO system at the same flight condition. Note that the upper plot, the $\underline{\lambda}$-plot, is identical to Figure 5.2. The frequency response of the SISO system is shown in Figure 5.4. The gain and phase margins from Figures 5.2 through 5.4 (computed using Equation 2.17) are tabulated in Table 5.1. It can be seen that the MIMO analysis, which yields the results in the $\underline{\sigma}_{min}$ column, is not much more conservative than the SISO $\underline{\lambda}$-plot analysis in this same column. Both analyses, however, yield conservative results when compared to classical results. This is because the $\underline{\sigma}_{min}$ must be interpreted as a boundary for gain, for phase, and for gain-phase combinations. One of these three is the worst case, but the same measure is used for all three, yielding conservative results. If, however the GM and P equations are applied <u>at the crossover frequencies of the Bode plot</u>, fair agreement is achieved between all methods (see Table 5.1). In fact, the eigenvalue plot is exactly right, except where it is equal to one. This indicates that although the conversion of $\underline{\sigma}_{min}$ to multiloop gain and phase margins is conservative, the $\underline{\sigma}$-plots and λ-plots themselves are giving a true picture of nearness to instability. For example, the SISO $\underline{\sigma}$-plot indicates that the worst case occurs at 20.4 rad/sec. On the frequency response at that frequency, a gain variation of 3.3 dB combined with a phase variation of 16.6 degrees will drive the system unstable. Plugging these two numbers into Equation (2.16) yields a SV of .396, exactly

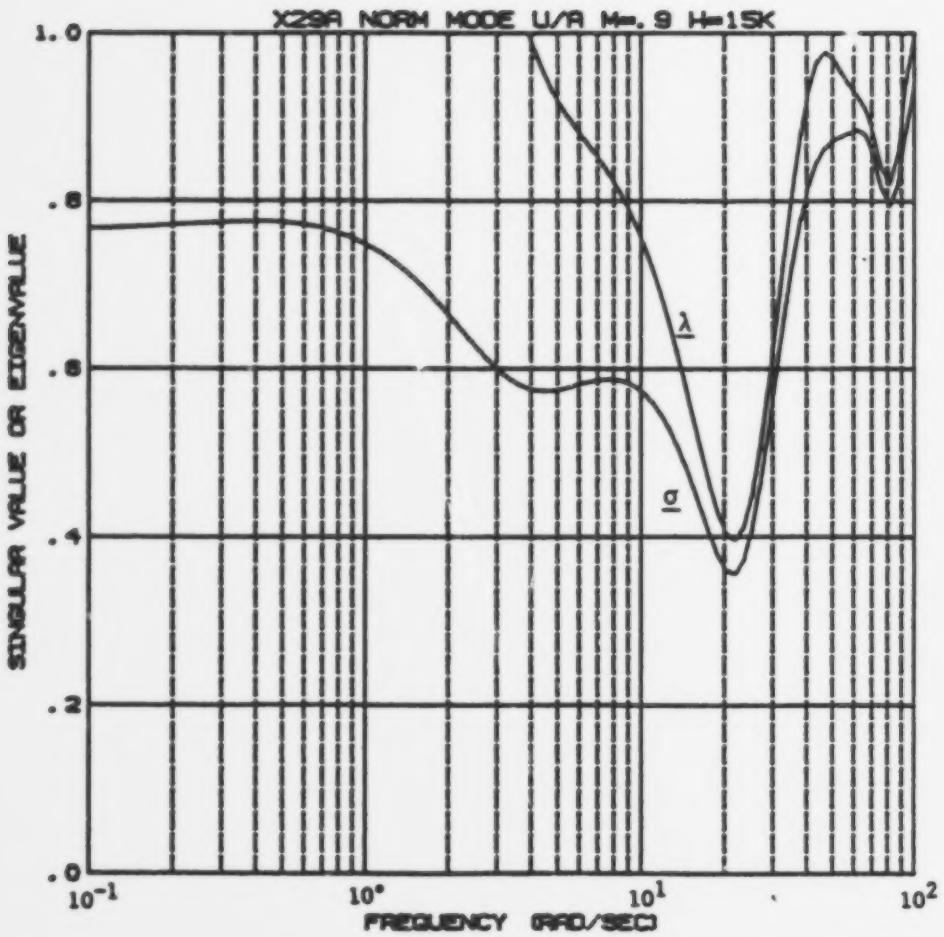


Figure 5.3: Singular value/eigenvalue plot for the X-29 longitudinal mode at M=.9 and H=15,000 ft, with the loop broken at point B in Figure 5.1.

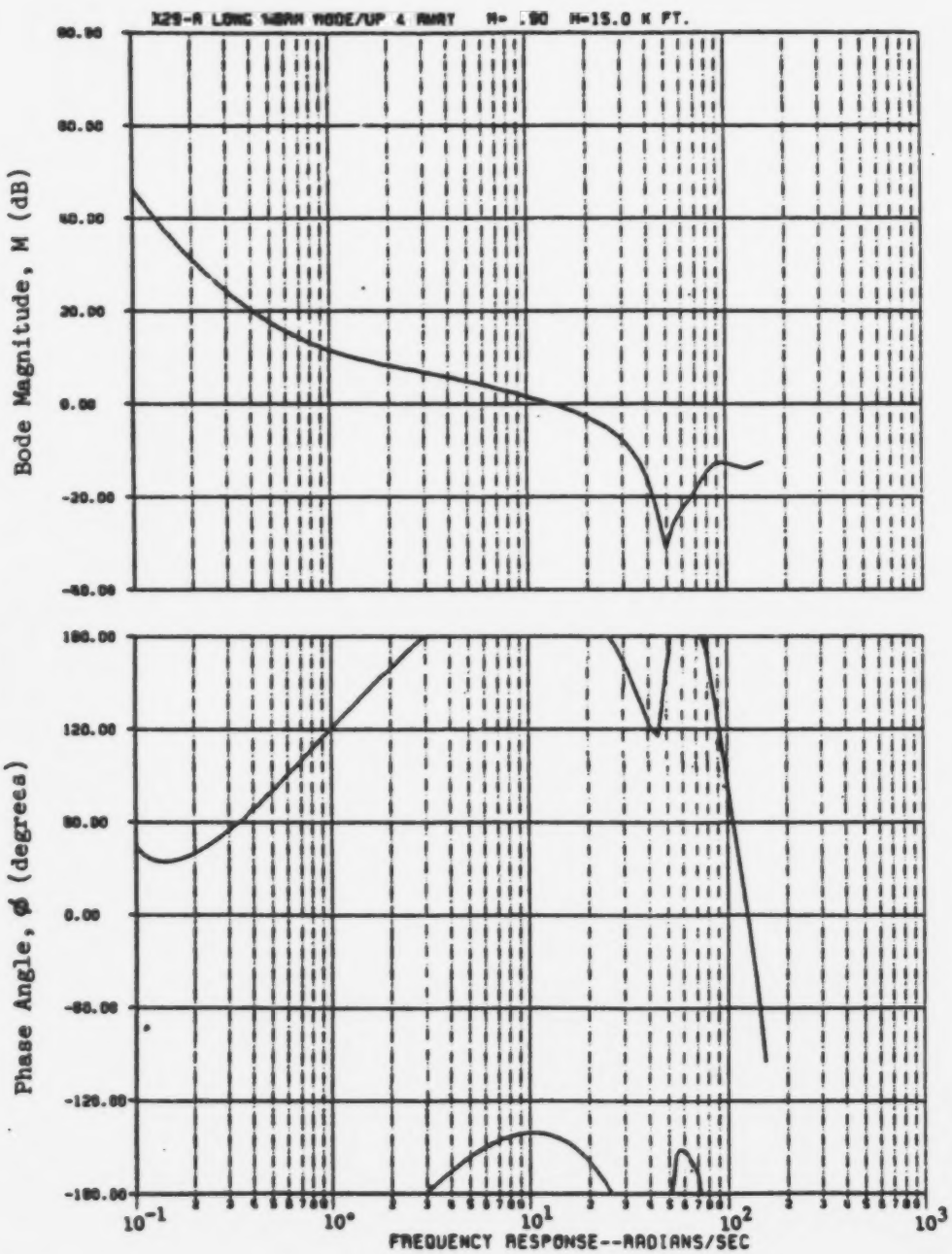


Figure 5.4: Frequency response for the X-29 longitudinal control system (loop broken at point A in figure 5.1).

Table 5.1: Gain and phase margin comparisons for the longitudinal X-29 at M=.9, H=15,000 ft

Frequency (rad/s):	3.0 (GM on Bode)	13.2 (PM on Bode)	21.9 ($\underline{\sigma}_{min}$ freq)	25.6 (GM on Bode)
From Bode Analysis				
GM, dB	-6.78	--	--	5.43
PM, deg	--	36.6	--	--
From MIMO $\underline{\sigma}$-Plot Analysis				
$\underline{\sigma}$	.59	.49	.36	.40
GM, dB	-4.0	--	-2.64 to 3.82	4.4
PM, deg	--	28.1	±20.5	--
From SISO $\underline{\sigma}$-Plot and MIMO λ plot Analysis				
$\underline{\sigma}$	1.0	.63	.40	.46
GM, dB	-6.02	--	-2.90 to 4.38	5.43
PM, deg	--	36.6	±22.8	--

as predicted by the $\underline{g}$-plot. Conversely, if one analyzes the SISO $\underline{g}$ at the frequency of a gain crossover, the worst case variation will be a phase margin (since gain is already at its critical value) and the $\underline{g}$-predicted phase margin will match exactly the actual (frequency response) PM. Gain margin behaves similarly at the phase crossover frequency. This method, of course, works only if the system is truely SISO. The point is that at the frequencies of the classical gain and phase margins, the matches are quite close, as indicated by Table 5.1. This fact will be true for multiloop systems also, although the predicted GM's and PM's will not match classical "one-loop-at-a-time" results exactly even at crossover frequencies.

5.1.2 Lateral Mode

Figure 5.5 shows the X-29 lateral mode block diagram. This system is truely MIMO, therefore classical analysis can yield over-optimistic results. See References [1] and [3] for examples of this phenomena, which was discussed in Section 2.1. In this section we will again be comparing classical and singular-value derived GM's and PM's but with the understanding that discrepencies may represent over-conservatism on the part of the MIMO analysis or over-optimism on the part of the classical analysis. Figures 5.6 is the $\underline{g}$-plot, and 5.7 and 5.8 are the frequency responses for 15,000 feet at a Mach number of .9; Table 5.2 tabulates the resulting gain and phase margins.

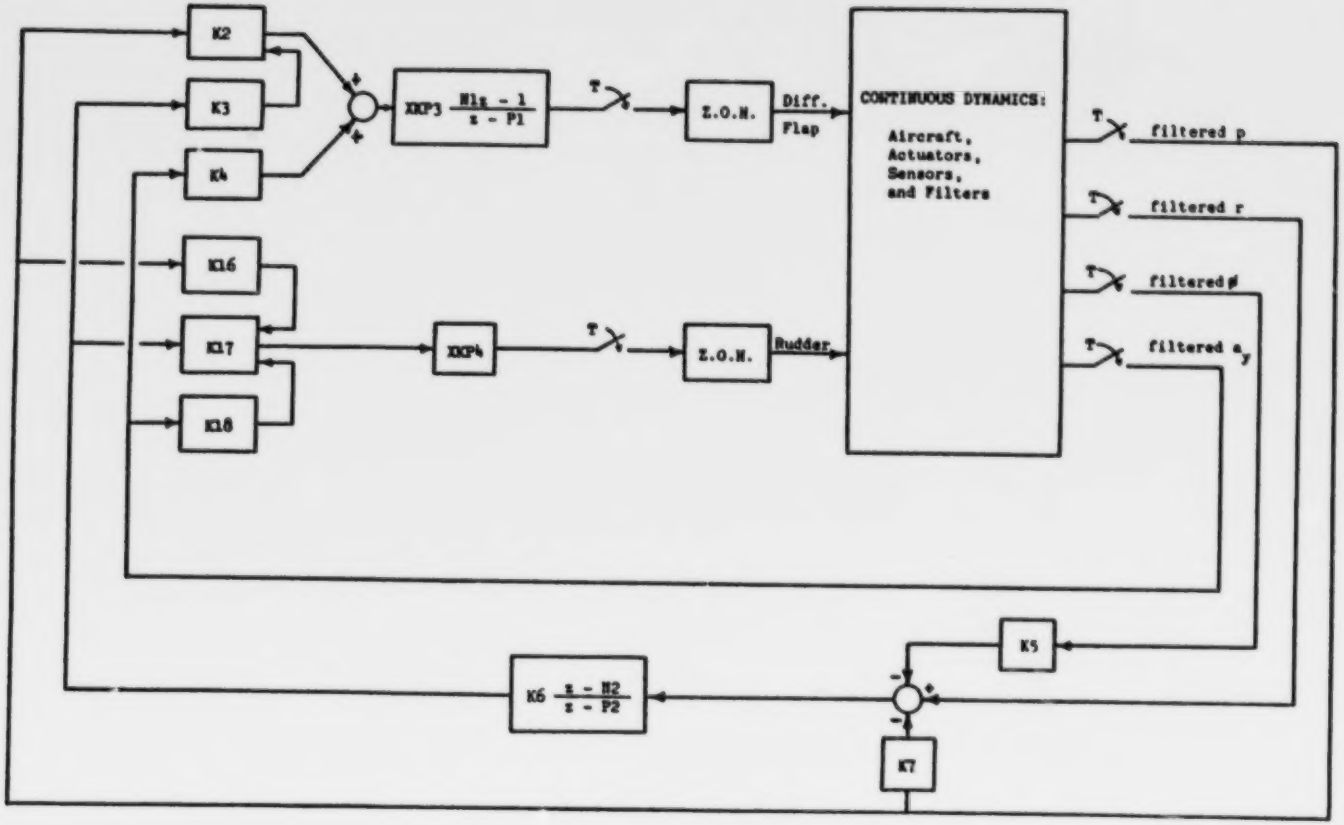


Figure 5.5 Simplified block diagram of the X-29 lateral-directional primary flight control system.

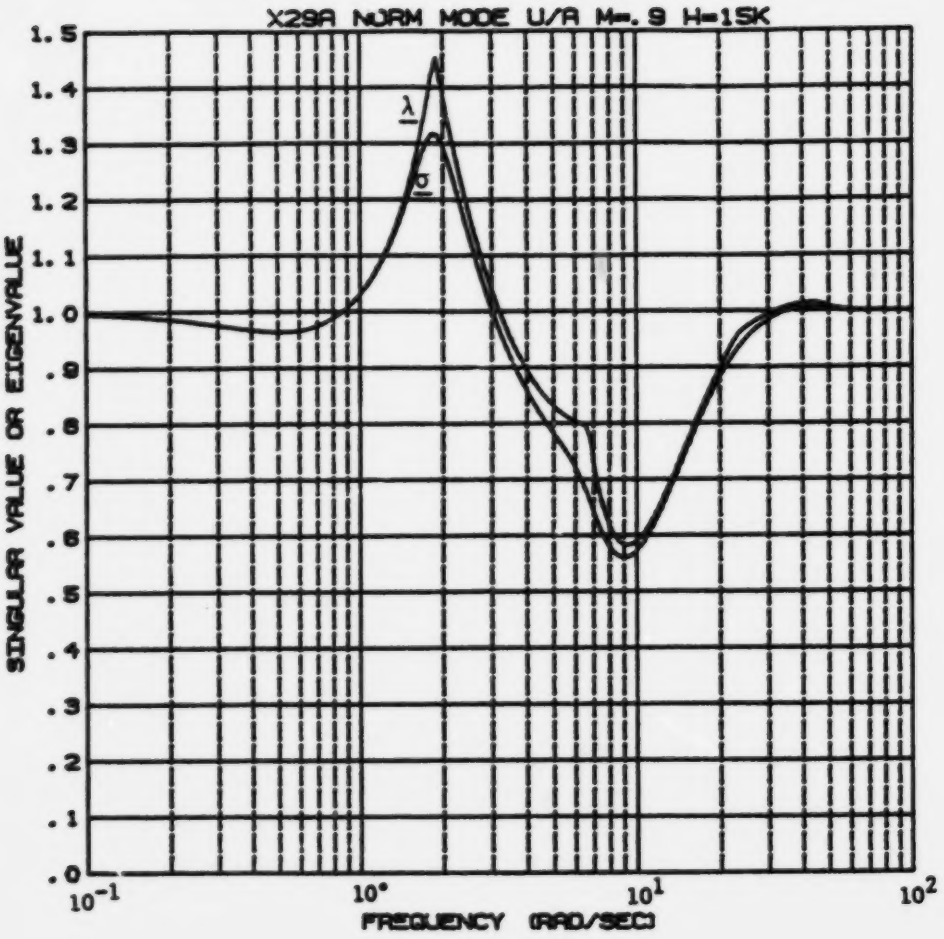


Figure 5.6: Singular value/eigenvalue plot for the X-29 lateral mode at M=.9 and H=15,000 ft

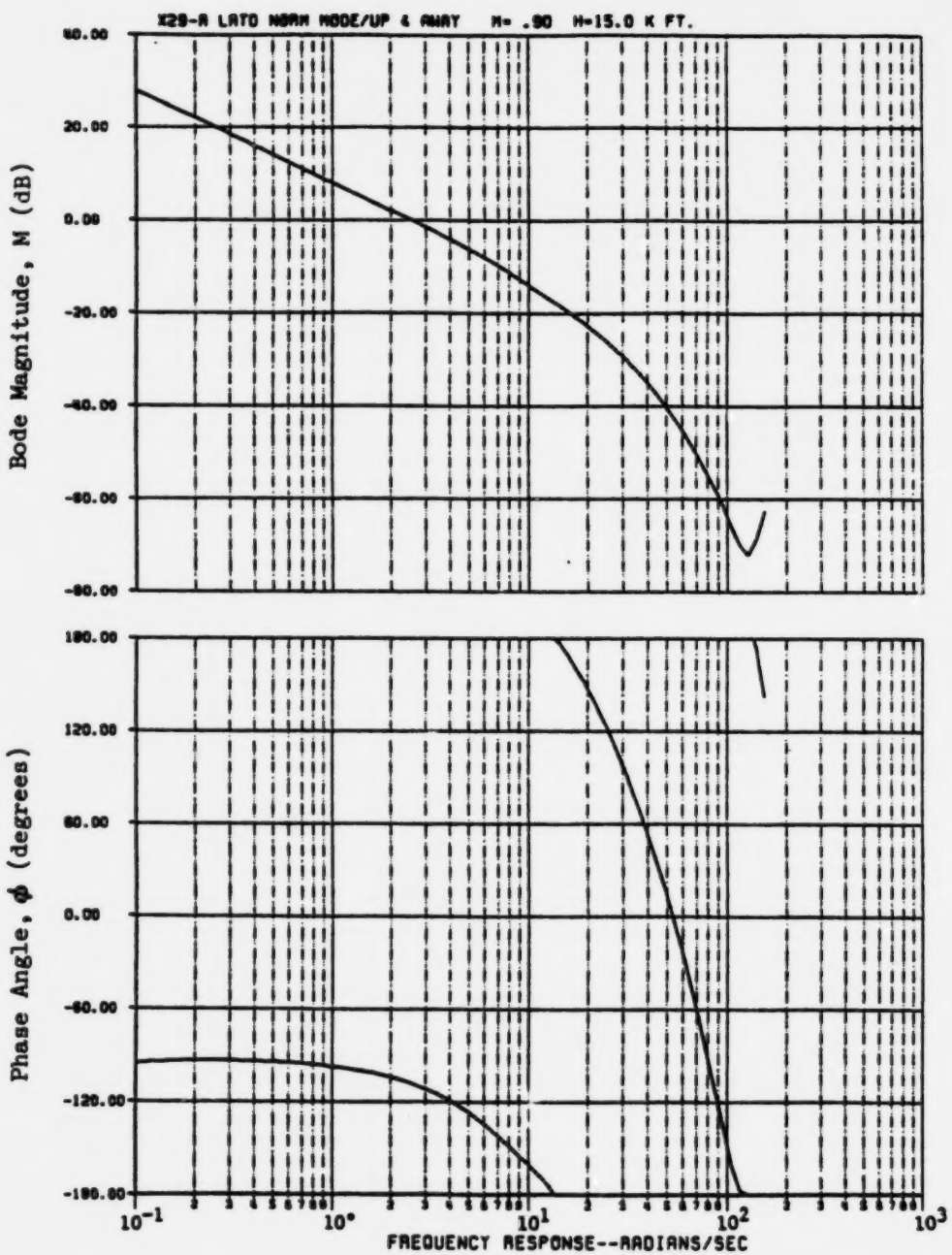


Figure 5.7: Aileron loop frequency response for the X-29 lateral-directional control system (loop broken at ZOH, rudder loop closed).

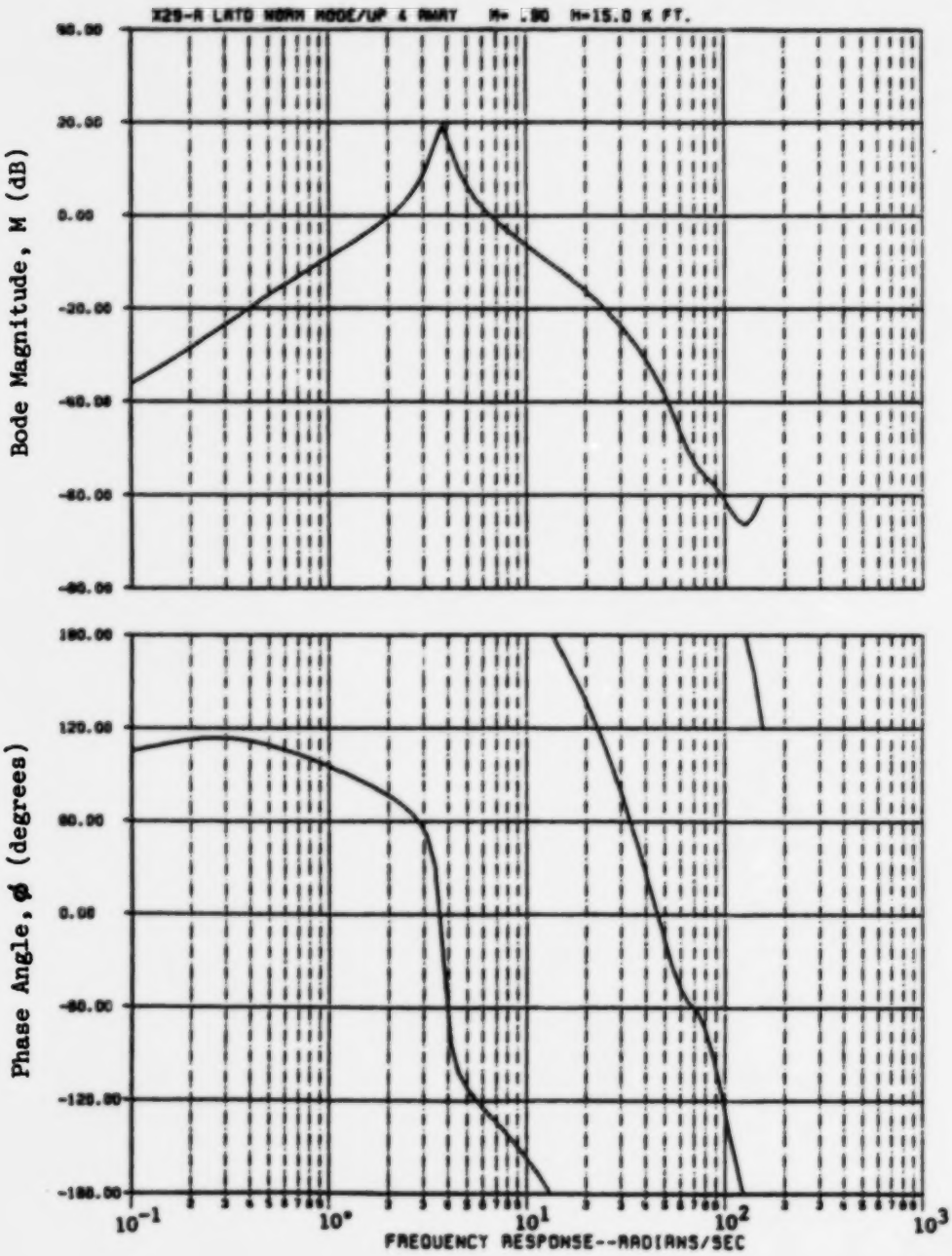


Figure 5.8: Rudder loop frequency response for the X-29 lateral-directional control system (loop broken at ZOH, aileron loop closed).

Table 5.2: Gain and phase margin comparisons for the lateral-directional X-29 at M=.9, H=15,000 ft

Frequency (rad/s):	2.4 (lateral PM on Bode)	6.5 (directional PM on Bode)	9.0 $\underline{\sigma}_{min}$ frequency	13.3 (directional GM on Bode)	13.8 (lateral GM on Bode)
From Bode Analysis					
GM, dB	—	—	—	10.2	17.6
PM, deg	73.1	52.3	—	—	—
From σ-Plot Analysis					
$\underline{\sigma}$	1.11	.66	.56	.68	.69
GM, dB	—	—	-3.86 to 7.13	9.79	10.2
PM, deg	67.2	38.7	±32.5	—	—
From σ-Plot Analysis					
$\underline{\lambda}$	1.15	.78	.58	.69	.71
GM, dB	—	—	-3.99 to 7.58	10.2	20.6
PM, deg	70.1	45.8	±33.9	—	—

Classical gain and phase margins were obtained by the method described in Section 2.1 and pictured in Figure 2.2 (one-loop-at-a-time). Again, it is apparent that the minimum of the sigma-plot represents the worst case variation, whether it be a combined change or a single change in gains or phases. For instance, at about 13-14 rad/sec, both the lateral and the directional Bode plots experience phase crossovers. But in this range (see Table 5.2), both the $\underline{\sigma}$-plot and the $\underline{\lambda}$-plot predict only the worst of the two gain margins, the directional GM, closely. They are very innacurate at predicting the lateral GM, because it is much higher and therefore not the worst case variation. Furthermore, notice that in this case $\underline{\sigma}_{min}$ occurs at 8.9 rad/sec, halfway between the directional GM frequency and the directional PM frequency. It predicts a GM and PM that are below all of the classical GM's and PM's, as one would expect. Again, because it is predicting this worst case, the lateral mode single-loop GM and PM are not matched accurately at all by the multiloop margins derived from $\underline{\sigma}_{min}$. Thus clarity of information has been lost because one number is being used to describe the entire situation. Of course, although singular value information is less accessible, it is more vital to stability of multiloop systems; it provides guarantees that cannot be gotten from frequency responses. An approach to interpreting singular values, based on this example and on the discussion in the previous section, is to look at the $\underline{\sigma}$-plot across the whole frequency range, and to realize that the different modes of the system are being represented

by relative minimums in the curve. By doing this a more realistic picture of the system's relative stability can be gained. Interactions between modes such as the one described above will also show up on the $\underline{\sigma}$-plots; this is information which is not available in the classical analysis.

5.2 SINGULAR-VALUE GRADIENTS APPLIED TO A REAL SYSTEM

Two sets of $\underline{\sigma}$-gradients were done for each of the control systems pictured in Figures 5.2 and 5.5.

The first set of $\underline{\sigma}$-gradients was done with respect to the aerodynamic and control power terms in the matrix model. These gradients would be useful in determining the robustness of the system, and in determining the information required from a flight test or other parameter identification program. They might also point out what types of design improvements are needed. For instance, if the control system is very sensitive to changes in C_{N_r} at the dutch roll frequency, a yaw damper loop may be necessary. Section 5.2.1 discusses a specific example.

The second set of $\underline{\sigma}$-gradients was done with respect to the control system block diagram elements. These gradients also indicate robustness, but the nominal values of the parameters in question are usually known to quite high accuracy. The usefulness of these gradients lies more in their ability to show where control system improvements might be available. An example is presented in Section 5.2.2, where the X-29 is used.

5.2.1 Aerodynamic Gradients

The aerodynamic parameters (both the dimensional force derivatives and the dimensional control power derivatives) appear, for the most part, as matrix elements in the $\bar{A}$ and $\bar{B}$ matrices, and not as combinations of these elements. As in Equation 4.14, these parameters are in the upper left-hand quadrant of the augmented $\bar{A}$ matrix, after control system, actuator, and sensor dynamics have been appended to the system. These are the parameters which are of primary interest to a sensitivity analysis, because they are known to the least accuracy. Table 5.3 shows the results of performing the $\underline{\sigma}$-gradient and data reduction analysis described in Section 4.2 on the aerodynamic parameters of the aircraft at a Mach number of .9 and an altitude of 15,000 feet. It is evident from this that very few of the aerodynamic parameters have a strong effect on the stability of the system. This indicates a well-designed system.

Table 5.3 indicates that M_α' and M_{DC}' (Pitching acceleration due to change in canard deflection) are the most important derivatives. This result agrees quite well with the results achieved by NASA-Dryden and Grumman personnel, who used classical techniques.

The singular value for 15,000 feet at a Mach number of .9, combined with the M_α' and M_{DC}' gradients, is plotted in Figure 5.9. The importance of frequency interaction is evident in this plot. Applying Equation (2.51) across the frequency range yields

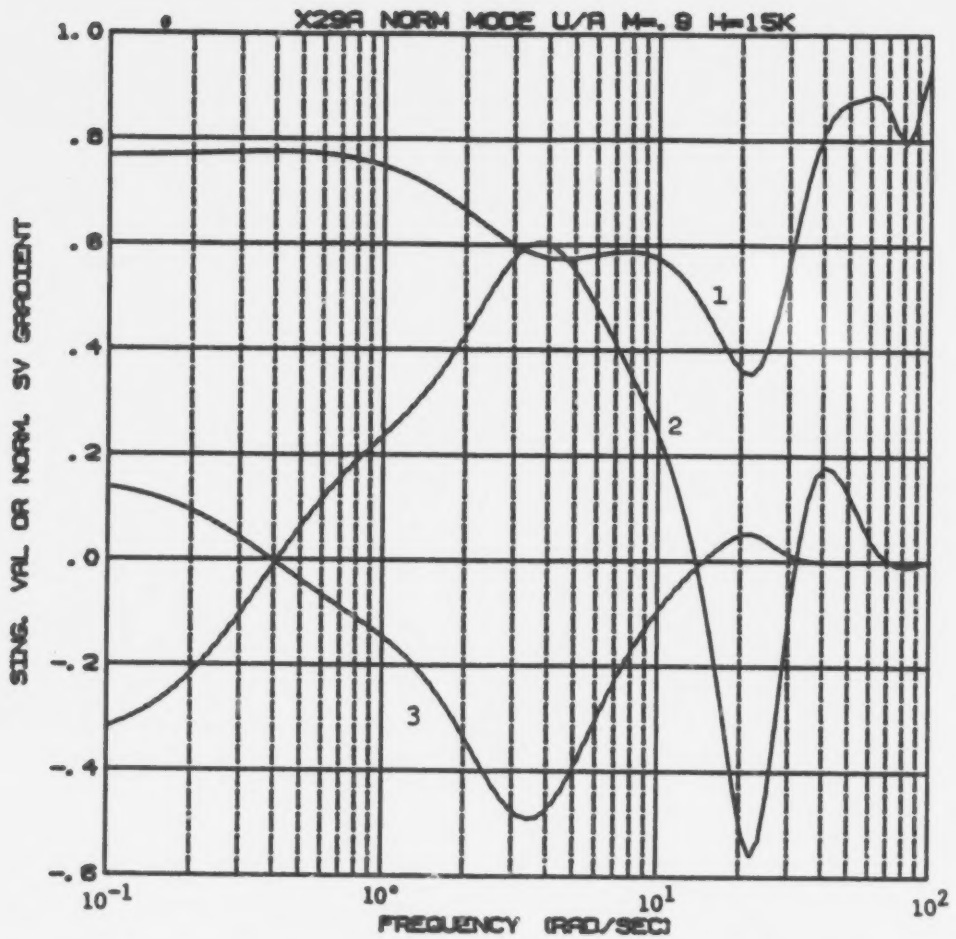


Figure 5.9: Plot of 1) singular value, 2) $\partial\underline{\sigma}/\partial M_{DC}'$, and 3) $\partial\underline{\sigma}/\partial M_{\alpha}'$ for the X-29 longitudinal mode at M=.9 and H=15,000 ft

Table 5.3: Sensitivity of the X-29 to changes in aerodynamic and control power derivatives, both longitudinal and lateral. Parameters are in descending order of importance (M=.9 and H=15,000 ft)

Parameter variations necessary to drive $\underline{\sigma}_{min}$ to .20 (see list of symbols for parameter definitions)

param.(p)	ω_{min}	$\underline{\sigma}(\omega_{min})$	$\partial\underline{\sigma}/(\partial p/p)$	Δp	$\Delta\underline{\sigma}$
longitudinal matrices:					
$B(3,1)\text{-}M_{DC_T}'$	22.5	.36	-.556	29%	-44%
$A(3,2)\text{-}M_{\alpha}$	3.47	.58	-.491	78%	-66%
$B(3,2)\text{-}M_{DSF}'$	22.5	.36	.184	-86%	-44%
$B(3,3)\text{-}M_{DST}'$	22.5	.36	.081	-196%	-44%
$A(2,2)\text{-}Z_{\alpha}$	1.98	.67	-.234	200%	-70%
$B(2,2)\text{-}Z_{DSF}'$	1.37	.72	.107	-484%	-72%
$A(3,3)\text{-}M_{q}$	15.5	.46	-.034	756%	-56%
$B(2,1)\text{-}Z_{DC}'$	1.65	.69	.0419	—	-71%
$B(2,3)\text{-}Z_{DST}'$	2.39	.63	-.0199	—	-68%
$B(1,1)\text{-}X_{DC_T}$	.211	.77	-.00191	—	-74%
$A(2,1)\text{-}Z_{u}$	.100	.77	.000965	—	-74%
$B(1,2)\text{-}X_{DSF}$	.211	.77	-.000616	—	-74%
$A(1,2)\text{-}X_{\alpha}$	.211	.77	.000535	—	-74%
$A(1,1)\text{-}X_{u}$	.100	.77	.000514	—	-74%
$A(3,1)\text{-}M_{u}'$	.100	.77	.000235	—	-74%
$B(1,3)\text{-}X_{DST}$	.145	.77	-.000090	—	-74%
$A(1,3)\text{-}X_{q}$	1.65	.69	.000008	—	-71%
lateral matrices:					
$B(3,2)\text{-}N_{DR}'$	10.6	.59	.309	-126%	-66%
$B(3,1)\text{-}N_{DDF}'$	7.33	.62	-.289	145%	-68%
$B(2,1)\text{-}L_{DDF}'$	1.98	1.30	.643	-171%	-85%
$A(2,2)\text{-}L_{p}'$	1.98	1.30	.448	-245%	-85%
$A(3,1)\text{-}N_{\beta}'$	1.65	1.27	-.343	311%	-84%
$A(3,2)\text{-}N_{p}'$	6.08	.72	-.144	359%	-72%
$A(2,1)\text{-}L_{\beta}'$	1.65	1.27	.155	-689%	-84%
$A(1,2)\text{-}Y_{p}'$	1.65	1.27	-.149	719%	-84%
$A(1,1)\text{-}Y_{\beta}'$	.940	1.01	-.104	782%	-80%
$B(2,2)\text{-}L_{DR}'$	1.65	1.27	.125	-857%	-84%
$A(1,2)\text{-}Y_{DR}'$	.940	1.01	-.0323	—	-80%
$A(3,3)\text{-}N_{r}'$	7.33	.62	-.00962	—	-68%
$A(2,3)\text{-}L_{r}'$	7.33	.62	.00612	—	-68%
$B(1,1)\text{-}Y_{DDF}'$	1.98	1.30	-.0156	—	-85%

Definitions:

param: Parameter for which gradient has been taken

ω_{min} : Frequency at which this parameter has maximum effect

$\underline{\sigma}(\omega_{min})$: Singular value at ω_{min}

$\partial\underline{\sigma}/\partial p/p)$: Singular-value gradient at ω_{min}

Δp: Percentage change in p necessary to drive $\underline{\sigma}_{min}$ to .20

$\Delta\underline{\sigma}$: Percentage change in $\underline{\sigma}$ necessary at ω_{min} to get to .20

NOTE: The derivatives Z_q' and Y_r' have been omitted from the analysis. They give misleading results because $Z_q'=1-Z_q/U1$ and $Y_r'=1-Y_r/U1$, so that very large changes in Z_q and Y_r are necessary to cause any significant effect on Z_q' and Y_r', respectively.

the plots of the variation in M_α' and M_{DC}' required to drive $\underline{g}$ to .2; these plots are shown in Figures 5.10 and 5.11 respectively. These plots show that the two most critical frequencies are ~4-5 rad/sec and 21 rad/sec. The interesting thing about Figure 5.11 is that it bounds the allowable variation in M_{DC}' in both the negative and the positive direction, giving the necessary accuracy of that parameter for the system to remain robust.

It is also interesting to note that if M_α' and M_{DC}' vary together in the same direction (which they are likely to do in the X-29, because the canard effectiveness is the primary contribution to both M_α' and M_{DC}'), the effects of their variations tend to cancel one another out. This result agrees with other sensitivity analyses, although in these analyses the cancellation is not as graphically evident as it is in Figure 5.9. The frequency nature of this cancellation is also evident; at the higher frequencies it does not occur; the M_α' gradient is large (negatively), while the M_{DC}' gradient is very small (positively).

Figure 5.9 and the above discussion suggest another way in which aerodynamic sensitivity information can be presented. Using aerodynamic concepts, one could set up equations which described how various combinations of parameters will probably change. For instance, if the lift-curve slope of the horizontal tail is predicted to be different than it actually is, the parameters which will be effected are

$$Z_\alpha,\ Z_{\dot{\alpha}},\ Z_q,\ M_\alpha,\ M_{\dot{\alpha}},\ \text{and}\ M_q$$

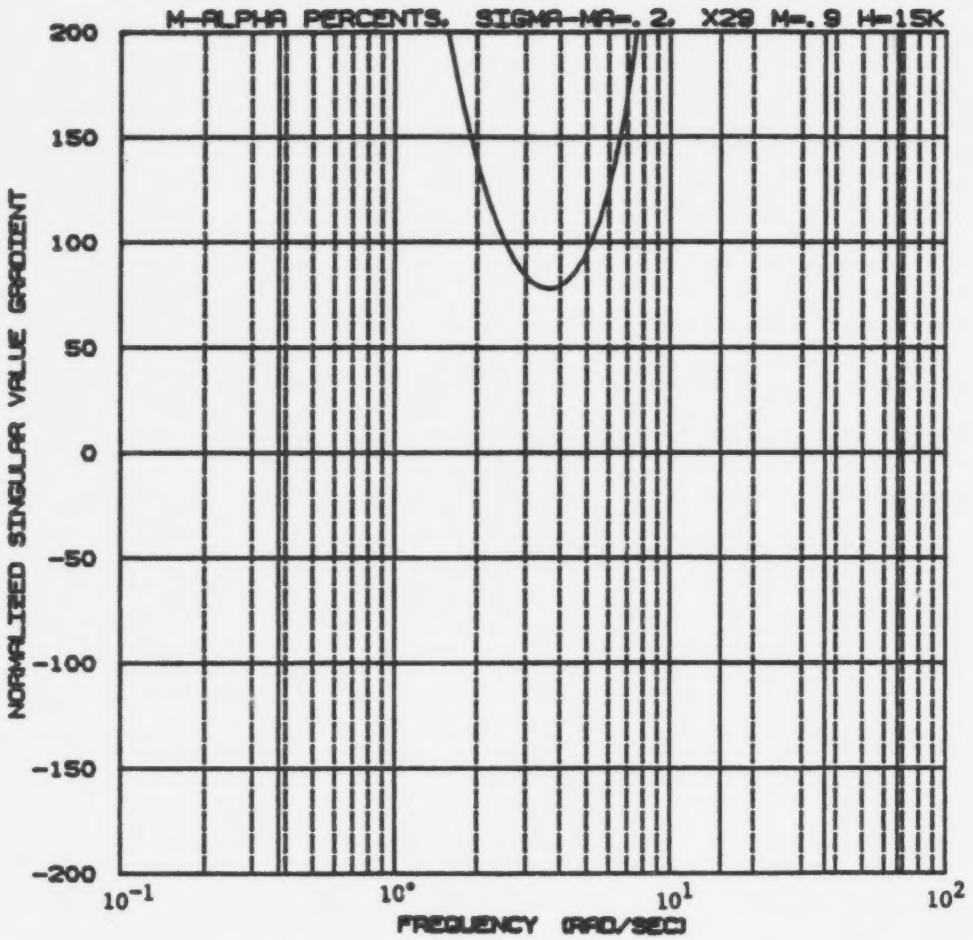


Figure 5.10: Plot of percent variation in M_{α}' required to drive $\underline{\sigma}$-plot minimum to .2 for the X-29 longitudinal mode at M=.9 H=15,000 ft

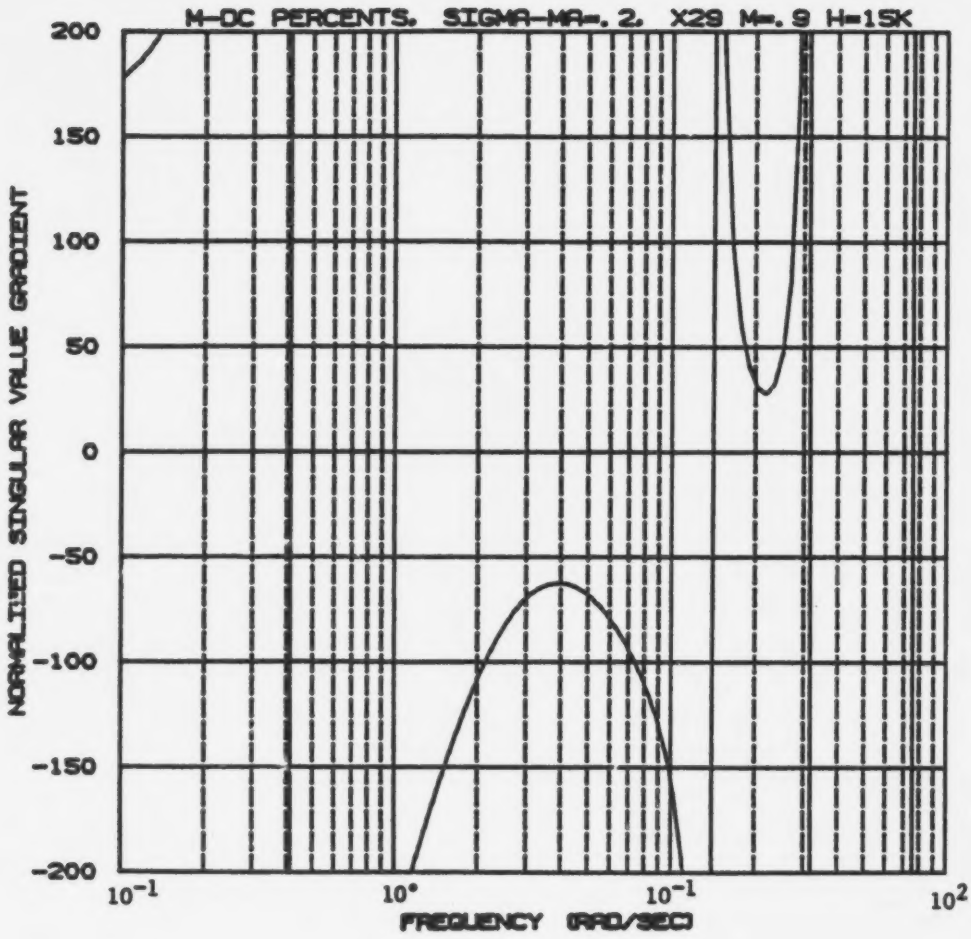


Figure 5.11: Plot of percent variation in M_{DC}' required to drive $\underline{\sigma}$-plot to .2 for the X-29 longitudinal mode at M=.9 and H=15,000 ft

These parameters will all vary when the lift curve slope of the horizontal tail changes. Thus a weighted sum of these parameters represents a parameter of the system which might change, and a singular-value gradient plot with respect to this weighted sum represents a very useful measure of the sensitivity of the system.

5.2.2 Control System Parameter Gradients

The control system parameters of interest to the designer are usually the constants that appear in the blocks of Figures 5.1 and 5.5, such as the compensator and filter break frequencies and damping ratios, and the feedback gains of the system. Using the partial-derivative expansion method described in Section 4.1, it is possible to get the gradients with respect to these parameters, even though they usually do not appear as single elements of the $\bar{A}$, $\bar{B}$, or $\bar{C}$ matrices. Table 5.4 gives the results of a sensitivity analysis on all the parameters in the longitudinal system. Table 5.5 gives similar results, for the parameters in the lateral control system. These results have been verified by perturbing some of the control system parameters, one by one, computing the resulting $\underline{\sigma}$-plot, and computing the derivatives numerically. Tables 5.4 and 5.5 are included to demonstrate the volume of information that can be sorted through by the analysis. Out of the 39 parameters in the lateral-directional and longitudinal control system that we analyzed, we were able to isolate the few that effect the stability of the control system most dramatically, and to order these from most to

Table 5.4: Sensitivity of the X-29 to changes in longitudinal control system parameters. Parameters are in descending order of importance (M=.9 and H=15,000 ft)

Parameter variations necessary to drive $\underline{\sigma}_{min}$ to .20

param.(p)	ω_{min}	$\underline{\sigma}(\omega_{min})$	$\partial\underline{\sigma}/(\partial p/p)$	Δp	$\Delta\underline{\sigma}$
P1	.100	.77	58.3	-1%	-74%
C3	18.6	.38	-1.58	12%	-48%
P2	6.08	.58	3.27	-12%	-66%
K3	18.6	.38	-1.20	15%	-48%
C2	18.6	.38	-1.05	17%	-48%
K2	18.6	.38	-.939	20%	-48%
GXA1	22.5	.36	.473	-34%	-44%
N1	22.5	.36	-.461	36%	-44%
G2	18.6	.38	.477	-39%	-48%
N2*N3	15.5	.46	-.674	39%	-57%
GXG2	22.5	.36	-.324	49%	-44%
C5	22.5	.36	.178	-90%	-44%
B2	22.5	.36	.148	-107%	-44%
C1	18.6	.38	-.151	122%	-48%
PST	.537	.77	-.460	125%	-74%
GS1	22.5	.36	.125	-128%	-44%
GF1	22.5	.36	.105	-152%	-44%
B1	18.6	.38	-.0895	205%	-48%
C4	22.5	.36	.0160	-992%	-44%
G1	6.08	.58	.0343	—	-66%
K1	4.18	.57	-.0274	—	-65%
KST	.647	.77	.0228	—	-74%
K4	22.5	.36	.000		-44%

Definitions:

param: Parameter for which gradient has been taken

ω_{min} : Frequency at which this parameter has maximum effect

$\underline{\sigma}(\omega_{min})$: Singular value at $\underline{\sigma}_{min}$

$\partial\underline{\sigma}/(\partial p/p)$: Singular-value gradient at ω_{min}

Δp: Percentage change in p necessary to drive $\underline{\sigma}_{min}$ to .20

$\Delta\underline{\sigma}$: Percentage change in $\underline{\sigma}$ necessary at ω_{min} to get to .20

Table 5.5: Sensitivity of the X-29 to changes in lateral control system parameters. Parameters are in decending order of importance (M=.9, H=15,000 ft)

Parameter variations necessary to drive $\underline{\sigma}_{min}$ to .20

param.(p)	ω_{min}	$\underline{\sigma}(\omega_{min})$	$\partial\underline{\sigma}/(\partial p/p)$	Δp	$\Delta\underline{\sigma}$
N2	.780	.99	13.2	-6%	-80%
P2	1.13	1.07	-13.3	7%	-81%
P1	2.39	1.16	12.9	-7%	-83%
N1	5.04	.78	-2.06	28%	-74%
K6	10.6	.59	-.386	101%	-66%
K17	10.6	.59	-.373	104%	-66%
XKP4	10.6	.59	-.373	104%	-66%
K2	2.39	1.16	-.553	174%	-83%
XKP3	2.39	1.16	.553	-174%	-83%
K7	7.33	.62	-.133	314%	-68%
K5	1.9	1.30	.0638	—	-85%
K3	8.83	.56	-.0178	—	-64%
K4	8.83	.56	.000	—	-64%
K16	8.83	.56	.000	—	-64%
K18	8.83	.56	.000	—	-64%

Definitions:

param: Parameter for which gradient has been taken

ω_{min}: Frequency at which this parameter has maximum effect

$\underline{\sigma}(\omega_{min})$: Singular value at ω_{min}

$\partial\underline{\sigma}/(\partial p/p)$: Singular-value gradient at ω_{min}

Δp: Percentage change in p necessary to drive $\underline{\sigma}_{min}$ to .20

$\Delta\underline{\sigma}$: Percentage change in $\underline{\sigma}$ necessary at ω_{min} to get to .20

least important. Of course, the plots and tabular data must be studied carefully to determine which parameters effect the system at the most important frequencies, what trade-offs result if a parameter changes, and the extent to which the linearity assumption is valid. But all the necessary data to make intelligent sensitivity judgements is now available. One example of the effect of a $\underline{g}$-gradient on a $\underline{g}$-plot will presented to demonstrate that the gradients are in fact correct. Figure 5.12 shows the $\underline{g}$-plot for the longitudinal control system at M=.9, H=15,000 feet, before and after a 50% variation in the feedback gain GXG2 (see Figure 5.1). The GXG2 gradient is also included on this plot, so that it is immediately clear that the prediction of the gradient is at least qualitatively correct. GXG2 is chosen because, although it is not the 'most important' parameter as measured by Table 5.4, its effect occurs at the most critical frequency of the system. Furthermore the trade-off which a variation in GXG2 imposes is quite acceptable; the minimum singular value can be increased by about .15, with a similar decrease occurring in the singular value at a higher frequency, where there is plenty of margin.

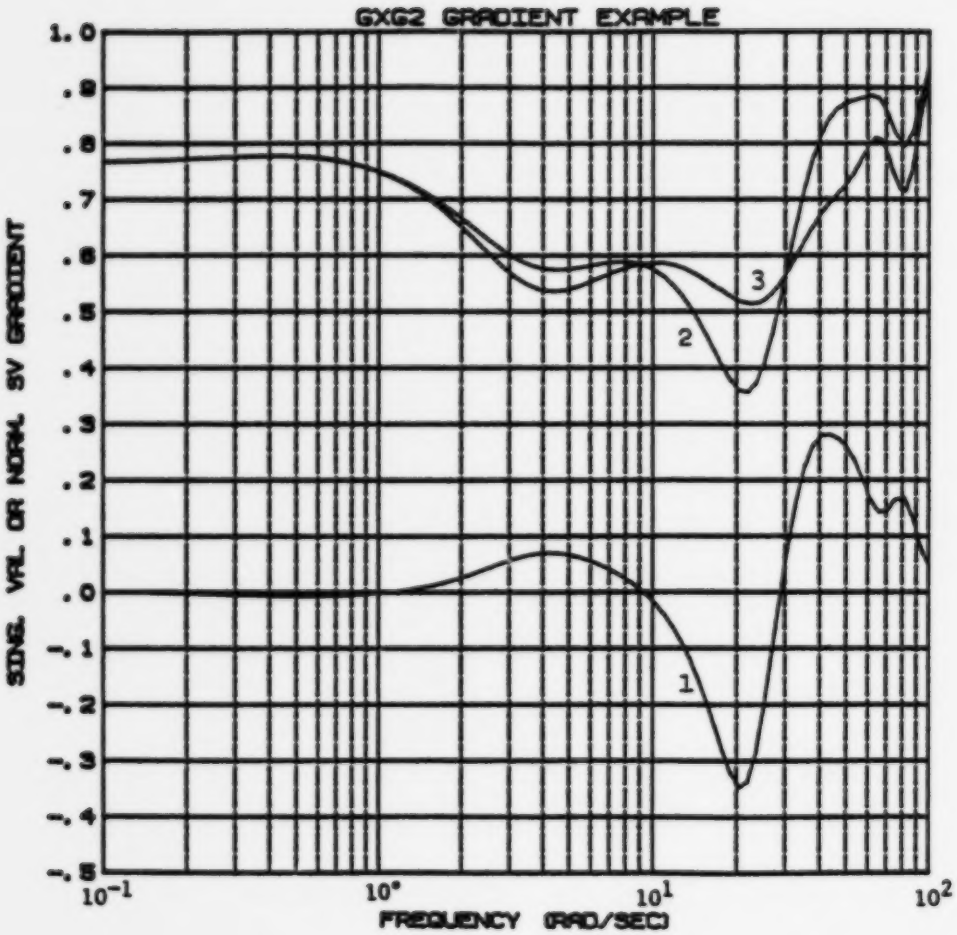


Figure 5.12: Effect of 1) $\partial \underline{g}/\partial GXG2$ on 2) $\underline{g}$ for the X-29 at M=.9 and H=15,000 ft. 3) is the plot which results when GXG2 is perturbed in the negative direction by 50%.

6. CONCLUSIONS AND RECOMMENDATIONS

In this paper it has been demonstrated that singular values and their gradients can be applied to systems which are of the complexity and size of problems in the real world. The extensions needed to analyze high-order digital systems do not make the analysis intractable. An important conclusion that resulted from this project was that the control system must be properly scaled if the singular values are to be useful as measures of sensitivity. The scaling techniques discussed in Section 2 are a vital first step in insuring that the analysis will give an accurate indication of the sensitivity of the control system to parameter changes.

Another important conclusion is that tabular results, although useful, are not usually adequate for this type of sensitivity analysis. Any method for tabulating the information in the $\underline{\sigma}$-plots and $\underline{\sigma}$-gradient plots must necessarily oversimplify both frequency interactions and the effects of nonlinearities. So, as with any new tool, a control system designer must work with the actual plots for a while, to get a feel for what they mean and what they do not mean. Given a proper understanding of the mathematical basis for singular values, the designer can get a lot of otherwise unavailable information from these plots.

Further research into the SVA might study a) the conservativeness of singular values as a robustness measure, b) the nonlinearity of the variation of the $\underline{\sigma}$-plot with parameter changes,

or c) the sensitivity of plots of singular values which are not the minimum.

The conservativeness of singular values is a current area of research. Many inroads have been made into reducing the unstructured and therefore conservative nature of singular values. Many of the less conservative robustness measures being studied are based on scaled singular values, which would lend themselves to the gradient technique. The methods for scaling the system to change the singular value plot presented in Section 2 yield an acceptable sensitivity measure; but non-real, non-diagonal scalings may yield better results. In any case, the effect of scaling the system on the gradient equations is of interest for future studies. The use of other robustness measures, such as structured singular values [21], in a sensitivity analysis is also a possible area for further reasearch.

The nonlinearity of singular values with respect to changes in parameters is a characteristic which, if properly understood, does not pose any real difficulty to the technique. The singular-value sensitivity analysis is a first-order analysis which is meant to pinpoint parameters which need further study. It is not meant to yield exact results; so nonlinearities, as long as they are not too great, do not invalidate the results. (See [23] for a discussion of the range of linearity of singular-value plots.) More work does need to be done, however, to quantify the level of nonlinearity that can be expected from singular-value plots when control system parameters change.

Finally, some way to account for the singular values which are not the minimum, and their gradients, must be implemented. This is necessary because $\underline{\sigma}$-plots might "cross-over" one another if their gradients are large enough. This is primarily a bookkeeping problem because it can be handled by simply performing the SVA on the second-smallest σ_1-plot instead of the smallest, and then on the third-smallest and so on. The parameter sensitivities can then be compared to get a full picture of the system's sensitivity. The implementation of this technique, and the organization of the increased amount of data that result, are the challenges which future studies might tackle. Alternate methods for accounting for larger singular values might also be studied.

REFERENCES

1. Herrera-Vaillard, A., Paduano, J. D., and Downing, D. R., "Development of a Sensitivity Analysis Technique for Multiloop Flight Control Systems," NASA CR-166619, October 1985.

2. Roskam, J., Airplane Flight Dynamics and Automatic Flight Controls, vols. I and II, Roskam Aviation and Engineering Corporation, Ottawa, KS., 1979.

3. Mukhopadhyay, V., and Newsom, J. R., "Application of Matrix Singular Value Properties for Evaluating Gain and Phase Margins of Multiloop Systems," AIAA Guidance and Control Conference, paper 83-1574, San Diego, CA, August 9-11, 1982.

4. Doyle, J. C., and Stein, G., "Multivariable Feedback Design: Concepts for a Classical/Modern Synthesis," IEEE Trans. Autom. Contr., vol. AC-26, No. 1, Feb. 1981.

5. Lehtomaki, N. A., Sandell, N. R., Jr., and Athans, M., "Robustness Results in Linear-Quadratic Gaussian Based Multivariable Control Designs," IEEE Trans. Autom. Contr., vol. AC-26, No. 1, Feb. 1981.

6. Yeh, H. H., Ridgely, Capt. B., and Banda, S. S., "Nonconservative Evaluation of Uniform Stability Margins of Multivariable Feedback Systems," AIAA Guidance and Control Conference, paper 84-1939, Seattle, Washington, August 1984.

7. Anderson, L. R., "Direct Design of Multivariable Control Systems through Singular Value Decomposition," AIAA Guidance and Control Conference, paper 83-2276, 1983.

8. Newsom, J. R., and Mukhopadhyay, V., "The Use of Singular Value Gradients and Optimization Techniques to Design Robust Controllers for Multiloop Systems," AIAA Guidance and Control Conference, paper 83-2191, 1983

9. Herrera-Vaillard, A., Paduano, J. D., and Downing, D. R., "Sensitivity Analysis of Automatic Flight Control Systems Using Singular Value Concepts," AIAA-85-1899, AIAA Guidance and Control Conference, August 1985.

10. Forsythe, G. E., and Moler, C. B., Computer Solution of Linear Algebraic Systems, Prentice-Hall, Englewood Cliffs, New Jersey, 1967.

11. Klema, V. C., and Laub, A. J., "The Singular Value Decomposition: Its Computation and Some Applications," IEEE Trans. on Autom. Contr., vol AC-25, No. 2, April 1980.

12. Applied Optimal Estimation, written by the Technical Staff, Analytic Sciences Corporation, edited by Arthur Gelb, MIT Press, Cambridge, Massachusetts, 1974.

13. Broussard, J. R., "Computation of Differential Equation Particular Solutions with Applications to Sigma Plots," IEEE Trans. on Autom. Contr., vol AC-29, No. 5, May 1984.

14. Maine, R. E., and Iliff, K. W., "Formulation and Implementation of a Practical Algorithm for Parameter Estimation with Process and Measurement Noise," SIAM Jour. Appl. Math., Vol. 41, No. 3, December 1981.

15. Kwakernaak, H., and Sivan, R., Linear Optimal Control Systems, Wiley-Interscience, New York, 1972.

16. Franklin, G. F., Powell, J. D., Digital Control of Dynamic Systems, Addison Wesley, Reading, Mass., 1981.

17. Ly, U., "Robustness Analysis of a Multiloop Flight Control System," AIAA-83-2189, AIAA Guidance and Control Conference, August 1983.

18. Mukhopadhyay, V., "Stability Robustness Improvement Using Constrained Optimization Techniques," AIAA-85-1931-CP, AIAA Guidance and Control Conference, August 1985.

19. Kouvaritakis, B. and Latchman, H., "Necessary and Sufficient Stability Criterion for Systems with Structured Uncertainties: the Major Principle Direction Alignment Principle," Int. J. of Control, 1985, Vol. 42, No. 3, pp. 575-598.

20. Safonov, M. G., "Stability Margins of Diagonally Perturbed Multivariable Feedback Systems," IEE PROC., Vol. 129, Pt. D, No. 6, November, 1982.

21. Doyle, J. C., Wall, J. E., and Stein, G., "Peformance and Robustness Analysis for Structured Uncertainty," Proceecings of the 21st IEEE Conference on Decision and Control, Vol. 2, December 1982.

22. Laub, A. J., "Efficient Multivariable Frequency Response Computations," IEEE Trans. on Autom. Contr., Vol AC-26, No. 2, April 1981.

23. Paduano, J. D., and Downing, D. R., "Sensitivity Analysis of High-Order Digital Flight Control Systems Using Singular-Value Concepts," AIAA paper 85-2084-CP, AIAA Guidance, Navigation and Control Conference, August 18-20, 1986.

APPENDIX A:

DERIVATIONS CONCERNING SINGULAR VALUES

Derivations considered too involved for the text are contained in this Appendix. Section A.1 contains proofs referenced in Section 2, Section A.2 contains proofs referenced in Section 3, and Section A.3 contains proofs from Section 4.

A.1 PROOFS FOR SECTION 2

The first derivation is of the basic theorem of singular value decomposition, which states that any matrix can be expressed as a real diagonal matrix combined with two unitary matrices. This derivation was referenced in Section 2.2. This proof is adapted from Reference [11].

Let A be any complex-valued m x n matrix of rank r. Then there exist complex unitary matrices U (m x m) and V (n x n) such that

$$A = USV^*, \tag{A.1}$$

where

$$S = \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix}$$

and $\Sigma=\text{diag}(\sigma_1,\sigma_2,\ldots\sigma_r)$ with

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$$

[For our special case, A, V, and U are all n x n; and, since A is of full rank, r=n and S=Σ]

Proof:

Since $A^*A \geq 0$ (nonnegative definite), the eigenvalue spectrum of A^*A, $\rho(A^*A) \subseteq [0,+\infty)$. Denoting $\rho(A^*A)$ by σ_i^2, $i=1,\ldots n$ we can arrange that $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_r > 0 = \sigma_{r+1} = \sigma_{r+2} = \ldots \sigma_n$. Let v_1, $v_2, \ldots v_n$ be a corresponding set of orthonormal eigenvectors and let

$$V_1 = (v_1, v_2, \ldots, v_r) \tag{A.2}$$

$$V_2 = (v_{r+1}, v_{r+2}, \ldots, v_n). \tag{A.3}$$

Then if $\Sigma = \text{diag}(\sigma_1, \sigma_2, \ldots \sigma_r)$, we have

$$A^*AV_1 = V_1\Sigma^2 \tag{A.4}$$

whence

$$\Sigma^{-1}V_1^*A^*AV_1\Sigma^{-1} = I. \tag{A.5}$$

Also $A^*AV_2 = V_2 \cdot 0$ so that

$$V_2^*A^*AV_2 = 0; \tag{A.6}$$

$$(AV_2)^*AV_2 = 0, \text{ and thus} \tag{A.7}$$

$$AV_2 = 0. \tag{A.8}$$

Let $U_1 = AV_1\Sigma^{-1}$. Then from (A.5) we have $U_1^*U_1 = I$; thus U_1 is unitary. Choose any U_2 such that $U = (U_1, U_2)$ is orthogonal. Then

$$U^*AV = \begin{bmatrix} U_1^*AV_1 & U_1^*AV_2 \\ U_2^*AV_1 & U_2^*AV_2 \end{bmatrix} \tag{A.9}$$

$$= \begin{bmatrix} \Sigma & 0 \\ U_2^*(U_1\Sigma) & 0 \end{bmatrix}$$

$$= \begin{bmatrix} \Sigma & 0 \\ 0 & 0, \end{bmatrix}$$

and so $A = USV^*$ as desired.

The next two proofs were also used in Section 2.2. The first is that the maximum and minimum singular values correspond to the Euclidean norms, which are the maximum and minimum change in length a matrix can produce as a transformation. The second is that

$$\bar{\sigma}(A^{-1}) = \frac{1}{\underline{\sigma}(A)} . \tag{A.10}$$

Both of these facts are proven in Reference [10] as follows:

Consider a matrix A as representing a linear transformation of one n-dimensional space X into a second such space Y. Thus $y = Ax$ is in Y and x in X. In representing the linear transformation by the matrix A we have assumed given orthogonal coordinate systems in both X and Y. Now consider an orthogonal change of coordinates in space X, so that the vector represented above by x obtains the new representation x' where $x = Vx'$. In the same way, by a different orthogonal coordinate change in Y, we obtain a new representation for y, namely y', where $y = Uy'$. Here both U and V are the matrices of [Equation A.1]. As a result of these changes of bases in X and Y the transformation originally represented by A obtains a new representation, which we will show to be S. We have

$$y' = U^*y = U^*A(Vx') = (U^*AV)x' = Sx'. \tag{A.11}$$

Thus $y' = Sx'$...

In the new orthogonal coordinate systems the transformation has a very simple representation. In terms of components we have

$$\begin{aligned} y'_1 &= \sigma_1 x'_1 \\ y'_2 &= \sigma_2 x'_2 \\ &\vdots \\ y'_r &= \sigma_r x'_r \\ y'_{r+1} &= 0 \\ &\vdots \\ y'_n &= 0. \end{aligned} \tag{A.12}$$

The transformation now merely maps the first coordinate axis of X onto the first coordinate axis of Y, with a magnification factor $\sigma_1 > 0$. It does the same for the 2nd, 3rd,...,r-th coordinate axes of X, with the respective magnification factors $\sigma_2,\ldots\sigma_r$. The (r+1)-th,...,nth coordinate axes of X are mapped onto the zero vector of Y.

From A.12 we can show that S maps the unit sphere $P = \{x' : \|x'\| = 1\}$ into an r-dimensional hyperellipsoid $E = SP$ of vectors y' such that

$$\frac{y'_1{}^2}{\sigma_1^2} + \ldots \frac{y'_r{}^2}{\sigma_r^2} = 1 \text{ and } y_{r+1} = \ldots = y_n = 0. \qquad (A.13)$$

One of the points of E furthest from the origin Θ is the point $(\sigma_1,0,0\ldots0)$. If $r < n$, then E contains the origin Θ. If $r = n$, then E does not contain the origin and one of the points of E closest to Θ is $(0,\ldots,0,\sigma_n)$. If $r<n$, then S and hence A are singular matrices. If $r=n$, S and A are nonsingular and have inverses; then directly from A.12 we see that

$$S^{-1} = \begin{bmatrix} \sigma_1^{-1} & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \sigma_n^{-1} \end{bmatrix}. \qquad (A.14)$$

Thus the singular values of A^{-1} are $\sigma_1^{-1},\ldots,\sigma_n^{-1}$.

From the discussion above and from the definition of $\|A\|$ as

$$\max_{x\neq 0} \frac{\|Ax\|}{\|x\|}$$

we see that

$$\|A\| = \|S\| = \sigma_1 \qquad (A.15)$$

(in other words, the singular values correspond to the matrix Euclidean norms, as was to be proved).

If r=n, then

$$\|A^{-1}\| = \|S^{-1}\| = \sigma_n^{-1} \qquad (A.16)$$

(in other words, $\bar{\sigma}(A^{-1}) = 1/\underline{\sigma}(A)$, as was also to be proved).

The last proof from Section 2.2 is taken from Reference [3]. This theorem allows the following stability criterion for the perturbed system return difference matrix: L must be smaller than the smallest matrix J for which

$$\underline{\sigma}[I + HGJ(j\omega)] = 0, \tag{A.17}$$

to be written as a criterion for the L matrix alone. To do this, we first rewrite [I + HGL] in a form that will allow G to be separated from L:

$$\begin{aligned} I + HGL &= (L^{-1} + HG)L \\ &= [(L^{-1}-I) + (I+HG)]L \\ &= [(L^{-1}-I)(I+HG)^{-1} + I](I+HG)L. \end{aligned} \tag{A.18}$$

Since (I+HG) and L are both nonsingular, (I+HGL) will be nonsingular if $[(L^{-1}-I)(I+HG)^{-1} + I]$ is nonsingular. This will be insured if [3]

$$\bar{\sigma}[(L^{-1}-I)(I+HG)^{-1}] < \bar{\sigma}[I] = 1. \tag{A.19}$$

but, according to [16],

$$\bar{\sigma}[(L^{-1}-I)(I+HG)^{-1}] < \bar{\sigma}[L^{-1}-I]\bar{\sigma}[I+HG^{-1}]. \tag{A.20}$$

So a sufficient condition for stability is

$$\bar{\sigma}[L^{-1}-I]\bar{\sigma}[(I+HG)^{-1}] < 1, \text{ or} \tag{A.21}$$

$$\bar{\sigma}[L^{-1}-I] < \bar{\sigma}^{-1}[(I+HG)^{-1}], \tag{A.22}$$

which translates, using Equation (A.10), to

$$\bar{\sigma}[L^{-1}-I] < \underline{\sigma}[I+HG], \text{ Q.E.D.} \tag{A.23}$$

A.2 PROOFS FOR SECTION 3

In Section 3.2, approximations for the derivatives of ϕ and Ψ are presented. The approximation for $\partial\Psi/\partial a_c$ is proven by applying it and comparing the results to the exact solution. The approximation for $\partial\phi/\partial a_c$ follows when the $\partial\Psi/\partial a_c$ equation is assumed.

The approximation for $\partial\Psi/\partial a_c$ is

$$\left.\frac{\partial\Psi}{\partial a_c}\right|_{\text{approx}} = \frac{1}{2}\,\Psi\,\frac{\partial A_c}{\partial a_c}\,\Psi\,. \tag{A.24}$$

If the Taylor series expansion for Ψ is substituted into this formula, the result is

$$\left.\frac{\partial\Psi}{\partial a_c}\right|_{\text{approx}} = \frac{1}{2}\left(T + \frac{A_c T^2}{2} + \ldots\right)\frac{\partial A_c}{\partial a_c}\left(T + \frac{A_c T^2}{2} + \ldots\right)$$

$$= \frac{1}{2}\left(T\,\frac{\partial A_c}{\partial a_c}\,T + \frac{A_c T^2}{2}\,\frac{\partial A_c}{\partial a_c}\,T + T\,\frac{\partial A_c}{\partial a_c}\,\frac{A_c T^2}{2} + \frac{A_c T^2}{2}\,\frac{\partial A_c}{\partial a_c}\,\frac{A_c T^2}{2} + \ldots\right)$$

$$= \left(\frac{1}{2}\,\frac{\partial A_c}{\partial a_c}\,T^2 + A_c\,\frac{\partial A_c}{\partial a_c}\,\frac{T^3}{4} + \frac{\partial A_c}{\partial a_c}\,A_c\,\frac{T^3}{4} + A_c\,\frac{\partial A_c}{\partial a_c}\,A_c\,\frac{T^4}{8} + \ldots\right.$$

(A.25)

the exact formula for $\partial\Psi/\partial a_c$ comes from taking the partial derivative of the Taylor series expansion, as follows:

$$\Psi = T + A_c\,\frac{T^2}{2} + A_c\,\frac{T^3}{3!} + \ldots \tag{A.26}$$

$$\frac{\partial\Psi}{\partial a_c} = \frac{\partial A_c}{\partial a_c}\,\frac{T^2}{2} + A_c\,\frac{\partial A_c}{\partial a_c}\,\frac{T^3}{3!} + \frac{\partial A_c}{\partial a_c}\,A_c\,\frac{T^3}{3!} + \ldots \tag{A.27}$$

Subtracting Equation (A.27) from (A.25) yields the error involved in the approximation.

$$\left.\frac{\partial \Psi}{\partial a_c}\right|_{\text{approx}} - \frac{\partial \Psi}{\partial a_c} = A_c \frac{\partial A_c}{\partial a_c} \frac{T^3}{12} + \frac{\partial A_c}{\partial a_c} A_c \frac{T^3}{12} + \frac{\partial A_c}{\partial a_c} A_c \frac{T^4}{12} + \cdots \tag{A.28}$$

so that the approximation is good up to the T^2 term of Ψ, with an error of order T^3.

To prove the approximation for $\partial\phi/\partial a_c$, we start with the following relation, which is obviously true if one studies Equations (3.9) and (3.10):

$$\phi = \Psi A_c + I; \tag{A.29}$$

$$\therefore \quad \frac{\partial \phi}{\partial a_c} = \frac{\partial \Psi}{\partial a_c} A_c + \Psi \frac{\partial A_c}{\partial a_c}$$

$$= \frac{1}{2} \Psi \frac{\partial A_c}{\partial a_c} \Psi A_c + \Psi \frac{\partial A_c}{\partial a_c}$$

$$= \frac{1}{2} \Psi \frac{\partial A_c}{\partial a_c} (\Psi A_c + 2I)$$

$$= \frac{1}{2} \Psi \frac{\partial A_c}{\partial a_c} (\phi + I). \tag{A.30}$$

A.3 PROOFS FOR SECTION 4

In Section 4.2.1, the computation of $(Is-A)^{-1}$ is mentioned as part of the algorithm to find the singular values. Computing $(Is-A)^{-1}$ for $s=j\omega$ at each frequency point takes by far the largest percentage of the total computation time. Thus it is necessary to

make this process as efficient as possible. After studying various alternatives, an algorithm presented by A. J. Laub [22] was used in the Dryden implementation of SVA. This algorithm is presented here, and extended to include digital systems.

The first step is to determine the similarity transformation that puts A in upper Hessenberg form. A Fortran subroutine which finds the necessary transformation is available in the ORACLS package, and yields T such that

$$A = THT^{-1}, \tag{A.31}$$

where H is an upper Hessenberg matrix, which means that the upper triangle and one subdiagonal of H are nonzero. The decomposition of A into H and T need only be done once, so adds very little computation time to the total process. $(Is-A)^{-1}$ can be written in terms of A and T as follows:

$$\begin{aligned}(Is-A)^{-1} &= [Is - (THT^{-1})]^{-1} \\ &= [T(T^{-1}s - HT^{-1})]^{-1} \\ &= [T(Is - H)T^{-1}]^{-1} \\ &= [(Is-H)T^{-1}]^{-1}T^{-1} \\ &= T(Is-H)^{-1}T^{-1},\end{aligned} \tag{A.32}$$

where the following fact has been used:

$$(AB)^{-1} = B^{-1}A^{-1}. \tag{A.33}$$

pre- and post- multiplying both sides of Equation A.33 by AB proves it to be true. To find $(Is-A)^{-1}$ at $s=j\omega$ we first solve

$$(Ij\omega-H)Z = T^{-1} \tag{A.34}$$

Then

$$(Ij\omega-A)^{-1} = TZ. \tag{A.35}$$

Suppose $Z=X+jY$ where $X,Y \in R^{nXn}$ (n is the dimension of A). Upon equating real and imaginary parts in Equation (A.34), we get the following 2·nth-order real system to determine X and Y:

$$\left[\begin{array}{c|c} -H & -\omega I \\ \hline \omega I & -H \end{array}\right] \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} T^{-1} \\ 0 \end{bmatrix} \tag{A.36}$$

Thus, $X=(1/\omega)HY$ and $Y=-\omega(\omega^2 I + H^2)^{-1}T^{-1}$. The matrix $(\omega^2 I + H^2)$ will be invertible if $(j\omega I-H)$ is invertible [22]. Note that $(\omega^2 I + H^2)$ is no longer upper Hessenberg, but is almost Hessenberg in the sense of having two, rather than one, nonzero subdiagonals. Its shape is wholly typified for n=5 by the matrix

$$\begin{array}{ccccc} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \end{array}$$

linear systems involving matrices of this type can be solved using approximately n^2 multiplications. We summarize the Hessenberg method using real arithmetic.

1) Reduce A to upper Hessenberg form H, find T and T^{-1}, and compute H^2; this step is only done once.

2) Solve $(\omega^2 I+H^2)Y=-\omega T^{-1}$ for Y.

3) Compute $X=(1/\omega)HY$ ($\omega\neq 0$; $(Is-A)^{-1}=-TH^{-1}T$ for $\omega=0$).

4) Compute $(Ij\omega - A)^{-1}=(TX) + j(TY)$

If the system is digital, we desire to compute $(Iz-A)^{-1}$, where

$$z = e^{j\omega T} = \cos(\omega T) + j\cdot\sin(\omega T). \tag{A.37}$$

For this case, Equation (A.36) can be written

$$\left[\begin{array}{c|c} -H-I\cdot\cos(\omega T) & -I\cdot\sin(\omega T) \\ \hline I\cdot\sin(\omega T) & -H-I\cdot\cos(\omega T) \end{array}\right]\left[\begin{array}{c} X \\ Y \end{array}\right] = \left[\begin{array}{c} T^{-1} \\ 0 \end{array}\right] \quad . \qquad \text{(A.38)}$$

If $\bar{\omega}$ is taken to be $\sin(\omega T)$ and $\bar{H}$ is taken to be $H-I\cdot\cos(\omega T)$, then Equation (A.38) can be written

$$\left[\begin{array}{c|c} -\bar{H} & -\bar{\omega}I \\ \hline \bar{\omega}I & -\bar{H} \end{array}\right]\left[\begin{array}{c} X \\ Y \end{array}\right] = \left[\begin{array}{c} T^{-1} \\ 0 \end{array}\right] \qquad \text{(A.39)}$$

and the derivation proceeds as above.

APPENDIX B:

DESCRIPTION OF THE X-29A ADVANCED TECHNOLOGY DEMONSTRATOR

The X-29A is currently being flight tested at NASA Ames-Dryden Flight Research Facility. This unique configuration is 35% unstable, so the verification of the flight control system is of primary importance to the success of the program. Thus very accurate information about both the aerodynamic characteristics and the control system characteristics is available. This information was used to construct the example runs presented in Section 5. This appendix presents the numbers necessary to describe the dynamics and feedback laws of the X-29 at a Mach number of .9 and an altitude of 15,000 ft.

Figure B.1 is a three-view of the X-29A. Table B.1 gives the dynamic matrices for the X-29 longitudinal linear model, for M=.9 and H=15,000 ft. The four states are velocity, angle of attack, pitch rate, and pitch attitude angle, in that order:

$$x^T = [\, u \quad \alpha \quad q \quad \Theta \,] \tag{B.1}$$

The longitudinal control variables are canard, symmetric trailing edge flaps, and strake flap:

$$u^T = [\, C \quad SF \quad ST \,] \tag{B.2}$$

The feedback laws of the primary (or "normal digital mode") longitudinal control system at this flight condition are modeled by

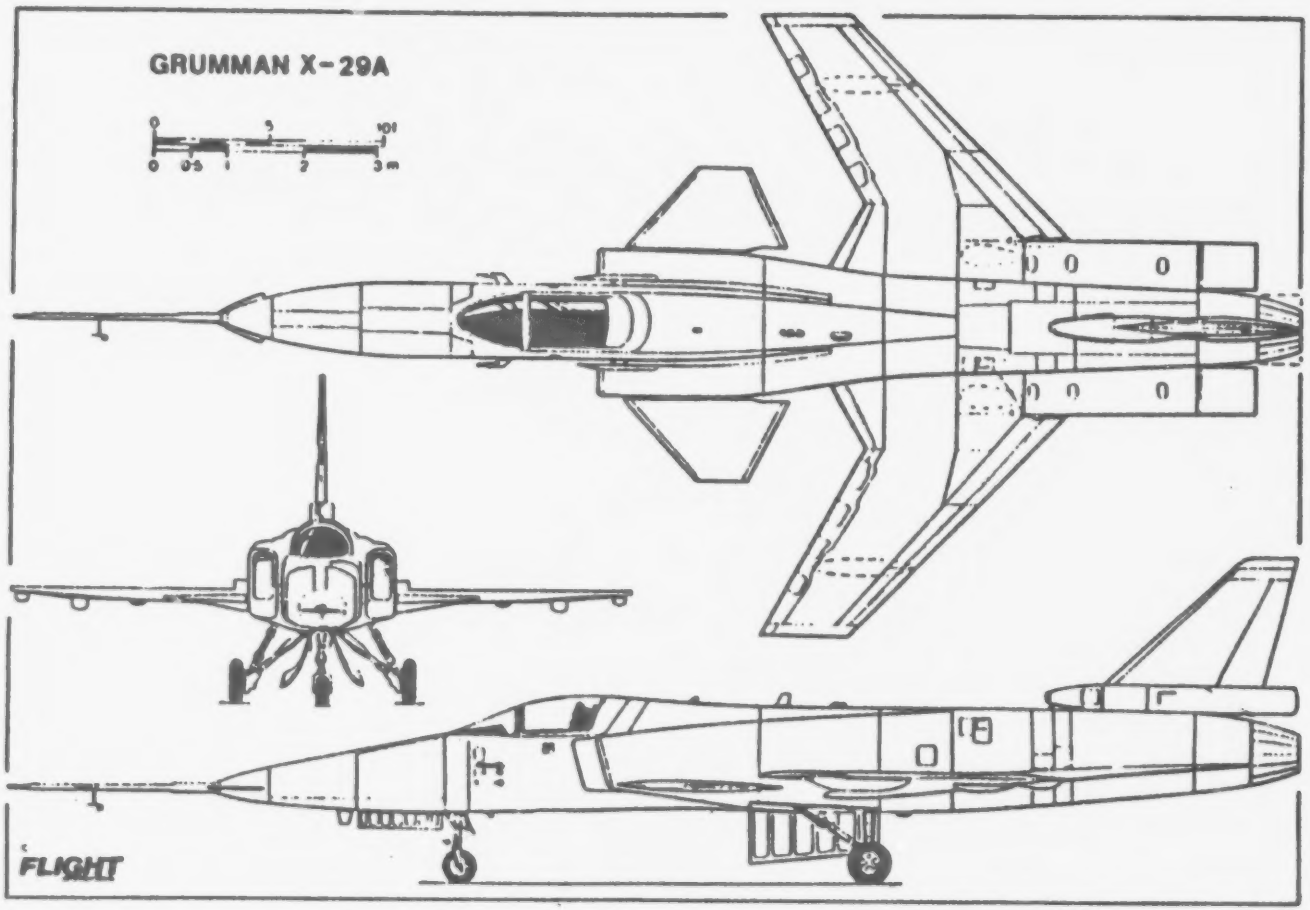


Figure B.1: X-29A Advanced Technology Demonstrator (June 1984 Flight International)

Table B.1 Longitudinal dynamic and control-power matrices for the X-29 at M= .9 and H=15,000 ft

A matrix:

	u(fps)	α(rad)	q(rad/s)	θ(rad)
$\dot{u}$	-.4368E-01	-.9045E+01	-.4514E+00	-.3213E+02
$\dot{\alpha}$	-.1485E-03	-.1791E+01	.9910E+00	.2522E-06
$\dot{q}$	-.6588E-03	.3578E+02	-.6981E+00	.0000E+00
$\dot{\theta}$	.0000E+00	.0000E+00	.1000E+01	.0000E+00

B matrix:

	DC(deg)	DSF(deg)	DDF(deg)
$\dot{u}$	-.1444E+00	.5415E-01	-.2530E-01
$\dot{\alpha}$	-.2043E-02	-.5604E-02	-.8410E-03
$\dot{q}$	.3376E+00	-.1909E+00	-.6556E-01
$\dot{\theta}$	.0000E+00	.0000E+00	.0000E+00

the diagram in Figure 5.1. The values for the control system parameters in this figure are given in Table B.2.

Table B.3 gives the dynamic matrices for the X-29 lateral linear model, for M=.9 and H=15,000 ft. The four states are sideslip angle, roll rate, yaw rate and bank angle:

$$x^T = [\ \beta \quad p \quad r \quad \phi\ ] \tag{B.3}$$

The lateral control variables are rudder and differential trailing edge flaps:

$$u^T = [\ R \quad DF\ ] \tag{B.4}$$

The feedback laws of the normal digital mode lateral control system at this flight condition are modeled by the diagram in Figure 5.5. The values for the control system parameters in this figure are given in Table B.4.

Table B.2 Gains and transfer function coefficients for the X-29 longitudinal control system - Figure 5.1 (M=.9 and H=15,000 ft)

K1	= 0.001	P1	= 1	C1	= 0.429
K2	= -0.1671	P2	= 1	C2	= -2.839
K3	= -0.1421			C3	= 3.843
K4	= 0.0			C4	= -0.04
KST	= 0.006183	PST	= 0.9756	C5	= -0.4
G1	= 1.428				
G2	= -3.565	N1	= 1		
GS1	= -0.70	B1	= 0.2308		
GF1	= -0.30	N2	= 1		
GXG2	= -5.568	N3	= 1		
GXA1	= -3.428	B2	= 0.2308		

Table B.3 Lateral-directional dynamic and control-power matrices for the X-29 at M=.9 and H=15,000 ft

A matrix:

	β(rad)	p(rad/s)	r(rad/s)	ϕ(rad)
$\dot{\beta}$	-.2872E+00	.5241E-01	-.9986E+00	.3371E-01
$\dot{p}$	-.4608E+02	-.4948E+01	.1651E+01	.0000E+00
$\dot{r}$	.1104E+02	-.1828E+00	-.8023E-01	.0000E+00
$\dot{\phi}$	.0000E+00	.1000E+01	.5248E-01	.0000E+00

B matrix:

	DDF(deg)	DR(deg)
$\dot{\beta}$	-.1465E-02	.1017E-02
$\dot{p}$	.1949E+01	.4671E+00
$\dot{r}$	.1257E+00	-.1203E+00
$\dot{\phi}$	.0000E+00	.0000E+00

Table B.4 Gains and transfer function coefficients for the X-29 lateral-directional control system - Figure 5.5 (M=.9, H=15,000 ft)

K2	= -1	XKP3	= 0.018
K3	= 3	XKP4	= 1
K4	= 0	N1	= 1.428
K5	= -0.03384	P1	= 1
K6	= 0.9938	N2	= 1
K7	= -0.05243	P2	= 0.9876
K16	= 0		
K17	= 0.59		
K18	= 0		

APPENDIX C:

USER'S GUIDE TO SVA

SVA stands for singular-value analysis. SVA is a program to compute the singular-value plot ($\underline{\sigma}$-plot) of the return difference matrix of a control system. The node at which the return difference matrix is computed (input or output) is controlled by the user's input matrices (see Sections 3.2 and 4.2.1). SVA also computes the "gradients", or partial derivatives, of the $\underline{\sigma}$-plot with respect to plant or controller parameters.

This appendix describes how to run SVA. It is broken up into five parts, four of which describe four ways in which SVA can be used:

C.1 Basic Definitions
C.2 Input Using a CONTROL Input File;
C.3 Input Using an Interactive Program Called PRESVA;
C.4 Input Directly From a Data File; and
C.5 Calling the Main Subroutine in SVA, Called SVANAL, Directly.

The way to read this guide is to study the definitions in Section C.1, and then skip to the section which corresponds to the way you wish to run the program. Each section contains the following information:

1) Discussion of the technique, including restrictions;
2) Definitions;
3) Input formats; and
4) Information necessary to run SVA on the ELXSI system.

C.1 BASIC DEFINITIONS

The definitions in this section apply to all the sections which follow.

C.1.1 Control System Description

The control system which is analyzed by SVA is of the form

$$\dot{x} = \bar{A}x + \bar{B}u \tag{C.1}$$

$$u = -\bar{C}x \tag{C.2}$$

for continuous systems, and

$$x_{k+1} = \bar{A}x_k + \bar{B}u_k \tag{C.3}$$

$$u_k = -\bar{C}x_k \tag{C.4}$$

for digital systems. Notice that SVA assumes negative feedback; you give it the C matrix to fit the above equations. Dimensions are

x : NSM x 1 ;
u : NC x 1 ;
$\bar{A}$: NSM x NSM ;
$\bar{B}$: NSM x NC ;
$\bar{C}$: NC x NSM .

For digital systems, $\bar{A}$ and $\bar{B}$ are of the following form:

$$\bar{A} = \left[\begin{array}{c|c} \phi(AC,T) & L12 \\ \hline L21 & L22 \end{array}\right] \tag{C.5}$$

$$\bar{B} = \left[\begin{array}{c} \Gamma(BC,T) \\ \hline G21 \end{array}\right] ,$$

where

$$\phi(AC,T) = I + AC\cdot T + \frac{AC^2T^2}{2!} + \frac{AC^3T^3}{3!} + \ldots ,$$

and $\Gamma(BC,T) = \int_0^T \phi(AC, \tau)d\tau \cdot BC.$

AC and BC describe the continuous subsystem of the overall control system, as illustrated in Figure 3.1.

T is the sample rate of the controller.

L12, L21, L22, and G21 are terms added in the z-plane, after discretization of the continuous dynamics, to account for the dynamics of the digital controller.

Dimensions for a digital control system are

x	:	NSM x 1 ;
u	:	NC x 1 ;
A	:	NSM x NSM ;
B	:	NSM x NC ;
C	:	NC x NSM ;
AC	:	NSU x NSU ;
BC	:	NSU x NCU .

C.1.2 Gradients

If you will not be computing gradients, you can skip this section. In later sections, if you do not want gradients, simply enter zero at the appropriate locations and no gradients will be computed.

A $\underline{\sigma}$-gradient with respect to some parameter p has the form

$$\frac{\partial \underline{\sigma}}{\partial p}(\omega) . \qquad (C.6)$$

It is a function of frequency, and describes the effect that the parameter p has on the $\underline{\sigma}$-plot. Since the $\underline{\sigma}$-plot is a measure of the relative stability of the system, a $\underline{\sigma}$-gradient shows the effect that varying p will have on the relative stability of the system. The $\underline{\sigma}$-plot will be effected by the parameter p in the following way:

$$\underline{\sigma}_{\text{perturbed}}(\omega) = \underline{\sigma}_{\text{baseline}}(\omega) + \frac{\partial \underline{\sigma}}{\partial p}(\omega) \cdot \Delta p. \qquad (C.7)$$

This equation illustrates an assumption made by SVA. SVA assumes that a first order Taylor series expansion approximates the behavior of the $\underline{\sigma}$-plot as p varies. This assumption will hold only within a certain range of Δp's.

To take out the effect of the relative size of various "p's", percent variations combined with <u>normalized</u> $\underline{\sigma}$-gradients are used by SVA. A percent variation is computed as

$$\frac{\Delta p}{|p|}(.100), \qquad (C.8)$$

and a normalized gradient as

$$\frac{\partial \underline{\sigma}}{(\partial p/p)} = \frac{\partial \underline{\sigma}}{\partial p} \cdot |p|. \qquad (C.9)$$

Thus Equation (C.7) becomes

$$\underline{\sigma}_{\text{perturbed}}(\omega) = \underline{\sigma}_{\text{nominal}}(\omega) + \frac{\partial \underline{\sigma}}{(\partial p/p)} \left(\frac{\Delta p}{|p|}\right). \qquad (C.10)$$

In this guide it is very important to distinguish between $\underline{\sigma}$-gradients and normalized $\underline{\sigma}$-gradients.

SVA allows normalized $\underline{\sigma}$-gradients to be obtained for any parameter p which is

1) An element of the matrices $\bar{A}$, $\bar{B}$, or $\bar{C}$ if the system is continuous.

2) An element of the matrices

$$AD = \left[\begin{array}{c|c} AC & L12 \\ \hline L21 & L22 \end{array}\right], \qquad (C.11)$$

$$BD = \left[\begin{array}{c} BC \\ \hline G21 \end{array}\right],$$

and $\bar{C}$ if the system is digital. Compare AD and BD to $\bar{A}$ and $\bar{B}$ in Equation (C.5); The AD and BD matrices are used instead of $\bar{A}$ and $\bar{B}$ because gradients with respect to ϕ and Γ are not of interest; see Section 3.

3) Any linear combination of elements of $\bar{A}$, $\bar{B}$, and $\bar{C}$ if the system is continuous.

4) Any linear combination of the elements of AD, BD, and $\bar{C}$ if the system is digital.

To avoid confusion, all gradients with respect to matrix elements (types 1 and 2) will be referred to as element-gradients. All gradients with respect to parameters which are linear combinations of matrix elements will be called partial-sum gradients. Partial-sum gradients are much harder to describe to SVA, so they should be avoided at first if possible. They are, however, quite necessary for some applications.

To get partial-sum gradient information, SVA calculates the following:

$$\frac{\partial \underline{\sigma}}{\partial a_{ij}}, \frac{\partial \underline{\sigma}}{\partial b_{ij}}, \text{ and } \frac{\partial \underline{\sigma}}{\partial c_{ij}}$$

as functions of frequency, where a_{ij}, b_{ij}, and c_{ij} are any elements of the $\bar{A}$, $\bar{B}$, or $\bar{C}$ matrices, respectively. SVA then computes the normalized $\underline{\sigma}$-gradient using partial derivative expansion equations of the following form:

$$\frac{\partial \underline{\sigma}}{\partial p} = \Sigma \frac{\partial a_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial a_{ij}} + \Sigma \frac{\partial b_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial b_{ij}} + \Sigma \frac{\partial c_{ij}}{\partial p} \frac{\partial \underline{\sigma}}{\partial c_{ij}}. \qquad (C.12)$$

The information which must be supplied to SVA for each p is

$$\frac{\partial a_{ij}}{\partial p}, \frac{\partial b_{ij}}{\partial p}, \text{ and } \frac{\partial c_{ij}}{\partial p}$$

for all i,j for which these derivatives are non-zero.

The way the partial-derivative expansion information is formatted in the SVA input file is only important if you plan to do direct data-file input; this is discussed in Section C.3. The kind of information which must be transferred to SVA is:

1) the number of terms in the partial-derivative expansion;
2) the original value of p;
3) the locations in $\bar{A}$, $\bar{B}$, and $\bar{C}$ of the elements a_{ij}, b_{ij}, and c_{ij} which make up the partial-derivative expansion; and
4) $\partial a_{ij}/\partial p$, $\partial b_{ij}/\partial p$, and $\partial c_{ij}/\partial p$ for each term in the expansion.

C.2 Using a CONTROL Input File to Run SVA

C.2.1 Discussion

The program CONSVA is simply a modified CONTROL. It runs in exactly the same way as CONTROL, except for a few modifications to the input file and the output filenames. These will be explained below.

CONSVA has been set up to run as a filter. It takes a CONTROL input file from $Stdin and outputs an SVA input file at $Stdout. It can be run using piping as explained in Section C.2.4. Control's normal output is sent to a scratch file called CONSVA.SCRATCH.

CONSVA serves two purposes. First, it allows control systems to be described using the block-diagram methods of CONTROL. It translates this information into the $\bar{A}$, $\bar{B}$, and $\bar{C}$ matrices required by SVA. For digital systems, it also provides the continuous matrices AC and BC. The second function of CONSVA is to generate all the information required by SVA to do normalized partial-sum gradients.

The following two sections provide more detail about what must be done to use SVA through CONTROL. Section C.2.1.1 discusses the form which the control system must take in order for CONSVA to work. Section C.2.1.2 discusses how the partial-sum gradients are computed.

C.2.1.1 Restrictions

The type of control system described by the CONTROL input file must conform to all the assumptions in SVA. This is especially important for digital systems. Figure C.1 shows the necessary format for the control system. This may look a little restrictive, but actually most of the restrictions represented graphically in Figure C.1 are already a part of CONTROL. The following additional restrictions are added to those imposed by CONTROL:

1. You must specify an open-loop analysis (SYSTEM=1), with the 'mixed' option (MIXED=1). (NOTE: Digital systems may also be analyzed with SYSTEM=2, but the gradients for these systems will not be correct.) The namelist variable FRPS is used to specify that singular values are desired. FRPS = 3, 4, or 5 specify 3 possible types of singular value analysis; these will be explained later.

2. You must create the block diagram, thin the y vector, and thin the u vector in such a way that the equation

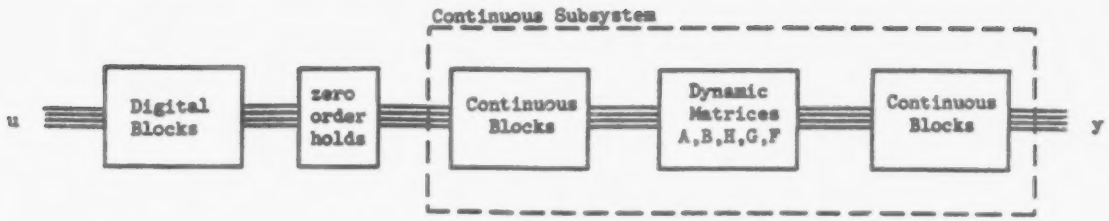
$$u = -y \tag{C.13}$$

will "close the loop" properly and create the closed-loop system that you're attempting to analyze. Thus both the dimensions of y and u and the sequence (or order) of there elements must match up exactly. Obviously Equation (C.13) also has implications about the signs of the various interconnections. For instance, if you have a CONTROL input file which correctly yields the closed-loop eigenvalues of the system, and you simply break the right loops to get the open-loop system represented in Figure C.1, it will be wrong because the SVA program assumes that there is a minus sign in the feedback path.

3. None of the variables being fed back (represented by CONTROL's y vector) can include control-position (u vector) elements. Very simply, this means that the augmented system resulting from this run through CONTROL must look like this:

$$\dot{x} = Ax + Bu, \tag{C.14}$$

For output node singular values:



NOTE: CONSVA can compute partial-derivative expansions only only for the structure shown below.

For input node singular values:

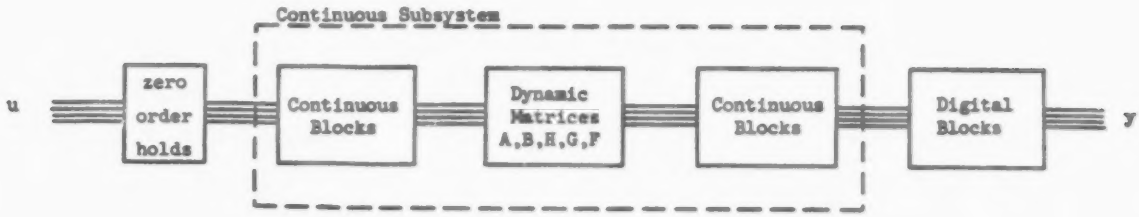


Figure C.1: Structure of control system for input to CONSVA

$$y = Hx. \qquad (C.15)$$

the F matrix must be zero, and setting u=-y and C=H must transform the above equations into the following representation of the system:

$$\dot{x} = Ax + Bu, \qquad (C.16)$$

$$u = -Cx. \qquad (C.17)$$

4. In CONSVA, the namelist variable IPT takes on a special meaning in addition to it's normal CONTROL meaning. If IPT=0, then no extra printout will be given from SVA. If IPT=1, a data echo will be printed out by SVA. This is very useful initially, to verify that everything's working. Finally, if IPT=2, the data echo for IPT=1 plus an echo of the matrices A, B, and C will be printed out.

5. The namelist variables IFREQ, FFREQ, DELFRQ, DIGITL, and DELT are passed on to SVA, so they should be specified with care. They mean the same things that they mean to CONTROL. The singular value analysis will proceed between IFREQ and FFREQ. If you don't want to have to wait forever for SVA to run, you had better specify DELFRQ = 1.2 or greater so that the number of points computed will be relatively small.

The following additional restrictions must be observed if gradients are desired. Restriction 7 arise because CONSVA cannot set up gradients for systems that yield the output-node return difference matrix if the system is digital.

6. The namelist variable CMAT must be = 0. This restriction applies to continuous as well as digital systems, whether the return difference matrix is at the input or the output node.

7. If you're analyzing a digital system you must break all loops at zero-order holds. This has several implications:

   a. Locations of the loop breaks are NOT arbitrary.

   b. The following namelist variables in CONTROL are restricted as follows:

      NUC = ZOH

NZTOU = 0

NYZTOK = 0

C.2.1.2 Partial-Sum Gradients

The parameters for which gradient plots are to be computed are specified at the very end of the input file to CONSVA. You just tack the information to be described onto the end of you're (otherwise sort-of standard) CONTROL file and you're in business. Sections C.2.2 and C.2.3 will describe the necessary variables and their formats.

CONSVA provides the information described in Section C.1.2, formatted and ready for SVA to read. It computes the derivatives

$$\frac{\partial a_{ij}}{\partial p}\ ,\ \frac{\partial b_{ij}}{\partial p}\ ,\ \text{and}\ \frac{\partial c_{ij}}{\partial p}$$

for each p specified by first introducing a Δp (small perturbation in p). It then recomputes $\bar{A}$, $\bar{B}$, and $\bar{C}$. Finally, it approximates the above derivatives for all i,j by computing

$$\frac{\Delta a_{ij}}{\Delta p}\ ,\ \frac{\Delta b_{ij}}{\Delta p}\ ,\ \text{and}\ \frac{\Delta c_{ij}}{\Delta p}\ .$$

The size of Δp is governed by the variable PERCENT. The default for Δp is 1% of p, but you may choose a different percentage.

C.2.2 Definitions

FRPS: CONTROL namelist parameter, used to indicate that singular value analysis is desired set to 3, 4, or 5, according to the directions in Section C.2.3.

NUMGRADS: Integer variable which tells CONSVA the number of partial-sum gradients desired.

LABEL: 10-character label which must be supplied for each gradient.

CODE: A one-character code variable which tells CONSVA the location of the parameter for which a partial-sum gradient is to be computed. CODE can take on the following values:

A : A matrix element (where A is the A matrix given in the input file to CONTROL)

B : B matrix element (where B is the B matrix given in the input file to CONTROL)

H : H '' '' '' H ''

F : F '' '' '' F ''

N : Numerator coefficients in block diagram (NUMER matrix in MIXED option)

D : Denominator coefficients in block diagram (DENOM matrix in MIXED option)

G : Block diagram gains (GAIN matrix in MIXED option)

IROW: Row in matrix indicated by CODE

ICOL: Column in matrix indicated by CODE (If CODE=G, ICOL is ignored and IROW is the desired element of the gain matrix)

PERCENT: Percent variation used to determine the partial derivative expansion. If not specified, PERCENT defaults to 1%. An entry of 10.0 would represents 10%. For parameters that have linear effects on matrix elements (which are the vast majority), PERCENT will have no effect.

C.2.3 Input Format

Set up a CONTROL input file, using the guidelines in the previous sections. In the namelist, set FRPS as follows:

FRPS = 3 for unscaled singular values

FRPS = 4 for singular values that are optimally scaled at each frequency ("structured" singular values)

FRPS = 5 for singular values that are scaled according to a user-input D matrix which is constant with frequency. (See Section 4.2.1).

Directly after the control data file information, a line containing NUMGRADS in I5 format is necessary. If NUMGRADS=0, no partial-sum information will be computed. If NUMGRADS≠0, two lines are needed for each of the NUMGRADS partial-sum gradients:

LINE 1: LABEL [in (A10) format]

LINE 2: CODE, IROW, ICOL [(4X, A1, 2I5) format]

'Stacked' cases of CONSVA can be run, and will result in stacked runs of SVA.

EXAMPLE: You've got a system with 5 states and 2 controls. To it you append 10 blocks to form a control system. You want to look at gradients for the following elements:

A(3,2), B(4,1), NUMER(10,1), DENOM(3,6), and GAIN(2).

The input you should append to your CONTROL input file is

```
    5
LABEL1
    A    3    2
LABEL2
    B    4    1
LABEL3
    N   10    1
LABEL4
```

```
    D    3    6
LABEL5
    G    2
```

Notice that

```
    A    6    3
```

would be an invalid entry because, even though the augmented matrix is larger than 5 X 5, only the elements of the original A can be specified. Other elements in the augmented A will be specified indirectly by using 'N', 'D', 'G', 'B', 'H', or 'F'.

If FRPS = 5, one additional line of data is required after the gradient information. This line must contain unformatted, real numbers, which represent the diagonal elements of the scaling matrix, D, as discussed in Section 4.2.1. This is the last line of each run when FRPS = 5. The dimension of D is determined by the number of thinned inputs and outputs in the system description.

C.2.4 To Run

BATCH:

Create the CONTROL input file. This file should conform to the specifications of Sections C.2.1, C.2.2, and C.2.3. Then type

```
BATCH +NOTIFY QUEUE=SLOW 'RUNCONSVA INPUT=CONTROLinputfile    &
                OUTPUT=SVAoutputfile PLOT=SVAplotfile'
```

When RUNCONSVA is used, CONSVA output is piped directly into SVA, so the SVA input file is lost. The output file will contain a short summary of the SVA run, including a table of the parameters specified using CODE, IROW, and ICOL, in their order of importance

as determined by the g-gradient technique explained in Section 4.2.3. The plot file will contain unformatted information used to run SVA interactively. To look at plots, type the following at the interactive prompt:

SVA

The ELXSI will respond with:

Type 'P' if this is a plot-only run:

your response should be 'p' or 'P', because the plotfile has already been created by RUNCONSVA, and this run of SVA is only to look at plots. The program will then prompt for the plotfile name. You should give it the name which you specified in the RUNCONSVA statement (at PLOT=). An interactive run of SVA begins at this point. SVA is a menu-driven program so no further explanation is given here.

If the program does not run properly, list the file CONSVA.SCRATCH and look at the last line in the file. This line will contain an error message if the input file to CONSVA is wrong. If there is no error message, check the SVAoutputfile for error messages.

An alternate way to run SVA is to specify an SVA input file name and give it to SVA directly. This can be done either in batch or interactively. First, type

CONTROLinputfile> CONSVA >SVAinputfile

The SVAinputfile can then be used as described in Section C.4.3.

You can also pipe the CONSVA output directly into SVA interactively, by typing

```
CONTROLinputfile> CONSVA | SVA >SVAoutputfile
```

The plotfile will go to a file named SVA.PLOTFILE in this case.

C.3 Input to SVA Using PRESVA

C.3.1 Discussion

If you have created the matrices $\bar{A}$, $\bar{B}$, and $\bar{C}$ (and AC and BC if the system is digital), you can run PRESVA to do all the rest of the work for you. PRESVA can also be used as a model for subroutines which do the same type of set-up, if you want to write a program that automatically interfaces some control design program to SVA. PRESVA is a very simple interactive program which allows you to describe element-gradients, partial-sum gradients, frequency ranges, dimensions, etc. From this information it creates an SVA input file. The code is self-documenting and relatively short, so it would be very simple to modify it to take information from a program such as CONTROL. PRESVA should not be considered a production tool unless gradients will not be computed, in which case it is probably the easiest way to go.

C.3.2 Definitions

The only term used by PRESVA whose definition will not be obvious is "coefficient". Partial-sum gradients are comprised of coefficients times element gradients, as follows:

$$C1 \cdot \frac{\partial \underline{\sigma}}{\partial a_{ij}} + \cdots + C2 \frac{\partial \underline{\sigma}}{\partial b_{ij}} + \cdots + C3 \frac{\partial \underline{\sigma}}{\partial c_{ij}} + \cdots . \qquad (C.18)$$

These coefficients are really just the $\partial a_{ij}/\partial p$, $\partial b_{ij}/\partial p$ and $\partial c_{ij}/\partial p$'s described in Section C.1.2.

C.3.3 Input Format

A data file must be provided which contains the matrices $\bar{A}$, $\bar{B}$, and $\bar{C}$, in that order if the system is continuous. If the system is digital the required matrices are AC, BC, $\bar{A}$, $\bar{B}$, and $\bar{C}$ in that order. No blank lines should appear in the file. The format for each matrix must be such that it can be read using the following FORTRAN code, assuming the dimension of the matrix is N x M :

```
              DO 10 I=1,N
10               READ(7,20) ( B(I,J) J=1,M )
20            FORMAT(3E24.12)
```

C.3.4 To Run

create the data file described above. Then simply type

PRESVA

at the interactive prompt. Answer all questions to create an SVA input file. Run this file using the instructions in Section C.4.4.

C.4 Data File Input to SVA

C.4.1 Discussion

Creating a data file for SVA from scratch is a little bit tricky if you are going to do partial-sum gradients. If you're not, it's probably the most straight-forward approach. Gradient definition in the SVA data file is a little confusing because it utilizes techniques designed specifically to save space in both the file and in the program, and to make the code more efficient. If you have already created a data file by the methods of Sections C.2 or C.3, skip directly to Section C.4.4 to see how to run SVA. If you are willing to get into some nitty-gritty, read on.

The difficulty that has been incorporated into both SVA and its input file is 'row packed' matrix referencing. Row packing is a way to number the matrix elements so that one number will give you the location of each element. A simple example is a 4 x 4 matrix; to reference any elements using row-packed notation, the elements would be numbered like this:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

So the 'row-packed location' of element (3,2) is 10. Naturally it is necessary to know the dimensions of the matrix to properly specify the location; for instance, the row-packed location of element (3,2) of a 4x2 matrix is no longer 10, but 6:

$$\begin{matrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{matrix}$$

SVA contains a further clever complication which is useful for describing the partial-derivative expansion of a parameter. the entire control system is referenced using 'super-packed' notation. 'Super-packed' can best be described by example. The elements of $\bar{A}$, $\bar{B}$, and $\bar{C}$ for a 3 state, 2 control system are numbered as follows:

$$\bar{A} = \begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix}$$

$$\bar{B} = \begin{matrix} 10 & 11 \\ 12 & 13 \\ 14 & 15 \end{matrix}$$

$$\bar{C} = \begin{matrix} 16 & 17 & 18 \end{matrix}$$

Obviously this is ridiculous if you're trying to do it by hand, especially if the matrices are large. That is why the self-documenting program PRESVA has been provided, to make the conversions from normal methods of indexing to the above method. For your application you may need to write a "filter" similar to CONSVA which utilizes the code in PRESVA to create an input file to SVA. Just get a listing of PRESVA to see how it works. The advantages to my indexing technique are very compact, pithy data files and smaller, more efficient, cleaner code.

Knowing what 'row-packed location' and 'super-packed location' mean complete the information required to describe the input file to SVA. In the following, we'll use the notation $\bar{A}$ [K] to denote the

row-packed location in a matrix, and $\bar{A},\bar{B},\bar{C}$ [K] to denote the super-packed location of an element in the $\bar{A}$, $\bar{B}$, or $\bar{C}$ matrix.

C.4.2 Definitions

Two integer arrays provide all the necessary information to specify element-gradients. MI is a four-element array specifying the Matrices of Interest.

I=1 refers to the $\bar{A}$ matrix
I=2 refers to the $\bar{B}$ matrix
I=3 refers to the $\bar{C}$ matrix
I=4 refers to the $\bar{P}$ matrix, which is described in References [1] and [8].

If MI(I) = 0, then matrix number I is not of interest - no gradients from that matrix are desired. If MI(I) = N, then there are N elements in matrix number I for which gradients are desired.

The locations of the elements for which gradients are to be computed are stored in a 4 row integer array called LES. Each row of LES corresponds to one of the 4 matrices as described above. For instance if MI(2)=4 then LES(2,J) for J=1,4 will contain the row-packed locations in $\bar{B}$ for which gradients are desired. SVA will compute the following set of gradients:

$$\frac{d\underline{\sigma}}{d\,\bar{A}\,[LES(1,J)]} \qquad J=1,MI(1);$$

$$\frac{d\underline{\sigma}}{d\,\bar{B}\,[LES(1,J)]} \qquad J=1,MI(2);$$

$$\frac{d\underline{\sigma}}{d\,\bar{C}\,[LES(1,J)]} \qquad J=1,MI(3); \quad \text{and}$$

$$\frac{d\underline{\sigma}}{d\,\bar{P}\,[LES(1,J)]} \qquad J=1,MI(4).$$

C.4.2.1 Definitions for Partial-Sum Gradients

The organization of the information for partial-sum gradients is more complicated than for element gradients, because of all the information that must be provided. It is stored in two arrays, a two-dimensional real array called FACS and a two-dimensional integer array called LOCS. Each row of LOCS corresponds to a row in FACS. The Nth row of both LOCS and FACS gives information about the Nth partial derivative expansion for which a $\underline{\sigma}$-gradient plot will be computed. We will describe the Nth row.

LOCS(N,1) contains the number of terms in the partial-derivative expansion of the gradient of interest, $\partial\bar{\sigma}/\partial p_N$. Because it is a counter, we'll call LOCS(N,1) K in the rest of this description.

LOCS(N,2) through LOCS(N,K+1) contain the super-packed locations of the elements a_{ij}, b_{ij}, and c_{ij} which make up the terms of the partial-derivative expansion of $\partial\underline{\sigma}/\partial p$.

FACS(N,1) contains the nominal value of the parameter p.

FACS(N,2) through FACS(N,K+1) contain the coefficients of the Nth partial derivative expansion. See Section C.1.2; these are the terms $\partial a_{ij}/\partial p$, $\partial b_{ij}/\partial p$, and $\partial c_{ij}/\partial p$.

using the notation we've set up for super-packing, the equation for the partial derivative expansion of gradient number N is simply

$$\frac{\partial \underline{\sigma}}{(\partial p_N / p_N)} = \left\{ \sum_{i=1}^{K} \text{FACS}(N,i+1) * \frac{\partial \underline{\sigma}}{\partial\{\bar{A},\bar{B},\bar{C}\}[\text{LOCS}(N,i+1)]} \right\} \text{FACS}(N,1) \qquad \text{(C.19)}$$

where K = LOCS(N,1).

C.4.3 Input Format

These are the contents of an input file to SVA (formats appear to the right of each line):

1: BATCH (A)
The word BATCH, in caps, fully left-justified, must appear as the first line of any data file.

2: TITLE (A)
Any 80-character title must be placed on line 2.

3: NSM, NC, NSU, NCU, NO, NS, M (7I5)

NSM = Number of states in the control system (Dimension of x vector)

NC = Number of controls in the system (Dimension of u vector)

NSU = Number of states in continuous portion of a sampled-data system (Dimension of AC; Ignored if DELT = 0.)

NCU = Number of controls in continuous portion of a sampled-data system (Number of columns in BC; Ignored if DELT = 0.)

NO, NS, M = Needed only if $\hat{P}$-gradients are to be computed; see References [1] and [8] for descriptions of these variables.

4: IPO, ISC, IG (3I5)

IPO specifies the various printout options:

IPO = 1 data echo; no plotfile output;

= 2 plotfile output; no data echo;

= 3 data echo and plotfile output;

= 4 same as 3 plus matrix echo.

ISC specifies the various scaling options:

= 1 unscaled singular values;

= 2 optimally scaled singular values ("structured," frequency dependent scaling);

= 3 user-input scaling matrix, D (see Section 4.2.1).

IG specifies the singluar value for which the analysis is done.

IG = 0 or 1 minimum singular value;

IG = 2 second-smallest singular value;

IG = n n^{th}-smallest singular value.

5: NP, FREQ1, FREQ2, DELT (I5, 3F10.5)

NP = Number of logarithmically spaced frequency points at which to compute the $\underline{\sigma}$-plot and $\underline{\sigma}$-gradient plots.

FREQ1 = Initial frequency for the plots.

FREQ2 = Final frequency for the plots.

DELT = Sample time for sampled-data systems. NOTE: DELT acts as a FLAG for digital systems as well as describing the sample time. So DELT must be = 0.0 if the system is continuous.

6: NUMPS (I5)

Number of partial-derivative expansions to be input

7 through ?:

For N = 1, NUMPS:

---

a: LABEL (A10)

Any 10-character label.

b: FACS(N,1), LOCS(N,1) (E10.6,I5)

FACS(N,1) = Nominal value of parameter p

LOCS(N,1) = Number of terms in the Nth partial-derivative expansion.

c: (LOCS(N,k+1), k=1,LOCS(N,1)) (15I5)

See Section C.4.2.1.

d: (FACS(N,k+1), k=1,FACS(N,1)) (6E12.6)

See Section C.4.2.1.

---

next 4 lines (I=1,4):

MI(I), (LES(I,J), J=1,MI(I)) (15I5)

See Section C.4.2.1 for a description of MI and LES.

Directly after all the above information, the matrices must appear in the following order (see Section C.3.3 for an explanation of the format:

AC matrix NSU x NSU (if DELT .ne. 0.0)
BC matrix NSU x NCU (if DELT .ne. 0.0)
A matrix NSM x NSM
B matrix NSM x NC
C matrix NC x NSM
D matrix diagonal elements (1 x NC, unformatted)
The D matrix diagonal elements are only needed if user-input scaling (ISC = 2) has been specified in line 4. Note that only the diagonal elements of the square matrix D are required; see Section 4.2.1 for details on scaling.

This completes the input for one run through SVA. Lines 2-end may be repeated as many times as desired to produced stacked runs for batch mode execution.

C.4.4 To Run

SVA can be run using any valid data file, whether it is a result of CONSVA (Section C.2), PRESVA (Section C.3), or any other method that conforms to the formats given in this section. The recommended way to use SVA is as follows:

1) Run SVA in batch mode. This is very simple to do, and for large systems it is necessary, because runtimes are relatively long.

2) Look at the SVA output file. This contains a quick summary of the run.

3) Run SVA interactively to get plots.

For small systems (NSM < about 20) steps 1 and 2 can be skipped.

SVA creates a "plotfile" during the first run (STEP 1). This file is then simply read in during subsequent runs (STEP 3) so that the time-consuming computations are only done once. Once the plotfile has been created, the input file is no longer needed. Step

3 can be repeated any number of times, so that plots can be viewed interactively over and over.

Step 1

To run SVA in batch mode, type

```
BATCH +NOTIFY QUEUE=SLOW 'RUNSVA INPUT=inputfile   &
                          OUTPUT=outputfile PLOT=plotfile'
```

SVA creates a temporary plotfile called SVA.PLOTFILE during batch execution. RUNSVA then renames this file to the plotfile name specified. List out the shellfile RUNSVA to see how this is done.

Step 2

The output file contains a short summary of the results of the run. The minimum of the $\underline{\sigma}$-plot is given, and translated into multiloop gain and phase margins. The normalized gradient information is then used to make a table of elements in their order of importance. This table gives the following information:

COLUMN 1:
A description of the matrix element. This is either in the form P(i,j) for element gradients (where P = $\underline{A}$, $\underline{B}$, or $\underline{C}$) or it is the label specified on input for partial-sum gradients.

COLUMN 2:
The frequency at which a change in the parameter has it's greatest effect.

COLUMN 3:
The minimum singular value of the return difference matrix at this frequency.

COLUMN 4:
The normalized singular value gradient, $((\partial\underline{\sigma}/\partial p) \cdot |p|$, at the same frequency.

COLUMN 5:
The percent variation in p ($\Delta p/|p|*100$) needed to drive the $\underline{\sigma}$-plot to $\underline{\sigma}$ = .2 has been chosen as a minimum singular value that would be undesirable. During interactive runs, this minimum can be changed.

Step 3

To run SVA interactively, simply type

SVA

at the interactive prompt. The first question asked by SVA is

Type 'P' if this is a plot-only run:

if a plotfile has been created as described in step 1, the answer to this question is 'p'. If you are running a small system and skipping the batch run, simple type carriage return. You will then be prompted for an input filename and a plotfile name. If you are doing a plotfile run, the input file is not necessary and will not be asked for.

In both plotfile runs and full computation runs, the next output you will receive is a menu, which is self-explanatory and will not be explained further here.

C.5 Calling SVA as a Subroutine.

For certain applications, calling SVA's main subroutine directly may be the most advantageous method. This subroutine is called SVANAL. The following description gives a complete calling sequence and guide to the input data required by SVANAL. Most of the information required is exactly the same as the information described in Section C.4; read Section C.4 before reading on. The subroutine SVANAL performs the following functions:

1. It does an optional echo check of all the input data.

2. It initializes all matrices and their dimensions. All matrices must be column-packed for use by this program, so SVANAL creates the column-packed form of the input matrices. It also creates the H matrix, which is used by Newsom and Mukhopadyay (see References [1] and [8]). The user supplies H matrix from the equation

$$z = Hx_s \qquad \text{(C.20)}$$

where z is the output vector of dimension NO and x_s is the state vector, of dimension NS. SVANAL creates

$$H = \left[\begin{array}{c|c} H & 0 \\ \hline 0 & I \end{array}\right] \quad \text{(NO+M x NS+M)} \qquad \text{(C.21)}$$

where H is the observer matrix defined in Equation (C.19) and I is an M x M identity matrix. This matrix is only created if gradients with respect to controller parameters (P matrix) are desired (see [1] and [8]). The user can also get a printout of H if desired.

3. It calls the subroutine SVCOMP, which handles the calculation of the singular values and their gradients.

4. It outputs singular value analysis information in matrix form. The following information is generated by SVCOMP (and the subroutines to which it is linked):

   a) Minimum singular value and minimum eigenvalue at each frequency.

b) Singular value gradients with respect to selected elements of the system and controller matrices, at each frequency

C.5.1 Calling Sequence

```
CALL SVANAL ( ABAR, MA, BBAR, MB, CBAR, MC, H, MH, D, NSM, NC,
              NO, NS, M, FREQ1, FREQ2, NP, MI, LES, NUMPS, LABELS,
              LOCS, FACS, DELT, AC, MAC, BC, MBC, NSU, NCU, IPO,
              ISC, LUN1, LUN2, NWKDIM, WK, IWK, SVMAT, MSV )
```

C.5.2 Input Arguments

ABAR - Variable dimensioned two-dimensional real array containing the augmented system matrix, $\bar{A}$. $\bar{A}$ has dimensions (NSM x NSM). For discrete systems, ABAR must contain the state transition matrix, ϕ.

MA - Maximum first dimension of the array ABAR as given in the DIMENSION statement of the calling program.

BBAR - Variable dimensioned two-dimensional real array containing the augmented control effectiveness matrix, $\bar{B}$. $\bar{B}$ has dimensions (NSM x NC). For discrete systems, BBAR must contain the discretized control power matrix, Γ.

MB - Maximum first dimension of the array BBAR as given in the DIMENSION statement of the calling program.

CBAR - Variable dimensioned two-dimensional real array containing the augmented feedback matrix, $\bar{C}$. $\bar{C}$ has dimensions (NC x NSM).

MC - Maximum first dimension of the array CBAR as given in the DIMENSION statement of the calling program.

H - Variable dimensioned two-dimensional real array containing the augmented observer matrix, H. This matrix is not required unless gradients with respect to parameters in $\hat{P}$ are desired, but the variable name must appear in the call statement.

MH - Maximum first dimension of the array H as given in the DIMENSION statement of the calling program.

D - NC x NC scaling matrix (for ISC = 2 option only).

NSM - Dimension of the augmented state-variable vector.

NC - Dimension of the control vector.

NO - Dimension of the output vector.

NS - State-vector dimension before augmentation.

M - Order of the controller-dynamics equation.

NOTE: The variables H, MH, NO, NS, and M must appear in the call statement to SVANAL, but they do not have to be initialized if gradients with respect to parameters in P are not desired. NO, NS, M and NC are consistent with the definitions given in Reference [8].

FREQ1 - Lowest frequency for the singular value analysis.

FREQ2 - Highest frequency for the singular value analysis.

NP - Number of frequencies between FREQ1 and FREQ2 inclusive where singular values, eigenvalues, and gradients are to be calculated.

MI - Integer vector whose dimension is 4. Each entry indicates the number of gradients desired from the $\bar{A}$, $\bar{B}$, $\bar{C}$, and $\hat{P}$ matrix respectively. This input argument is described in more detail in Section C.4.2.

LES - Integer array whose first dimension must be 4. Second dimension must be $\max_k(MI(k))$. Entry (k,i) indicates the row-packed location (counting across the rows) in the k-th matrix ($\bar{A}$, $\bar{B}$, $\bar{C}$, or $\hat{P}$) of the parameter with respect to which a gradient plot is desired. This input argument is described in more detail in Section C.4.

NUMPS - Number of desired partial-sum gradients.

LABELS - CHARACTER*10 array with 25 elements. The first NUMPS elements should contain the labels for the partial-sum gradients.

LOCS - Integer array of dimension (25 x 100). Each row contains information on one of the NUMPS partial-sum gradients. The first column tells how many coefficients are in the partial-derivative expansion for the gradient. Each column after the first contains the super-packed location of the elements of $\bar{A}$, $\bar{B}$, and $\bar{C}$, that are in the expansion. See Section C.4.2 for more detail.

FACS - Real array of dimension (25 x 100). Each row contains information on one of the NUMPS partial-sum gradients. The first column gives the nominal value for the parameter with respect to which the gradient is being taken. Each additional column gives the coefficents of the partial derivative expansion equation. See Section C.4 for more detail.

DELT - Sampling interval for sampled-data systems. This variable must be set to zero for continuous systems.

AC - Variable dimensioned two-dimensional real array containing the continuous system matrix. AC has dimensions (NSU x NSU). AC is only used when digital systems are being analyzed.

MAC - Maximum first dimension of the array AC as given in the DIMENSION statement of the calling program.

BC - Variable dimensioned two-dimensional real array containing the continuous control power matrix. BC has dimensions (NSU x NCU). BC is only used when digital systems are being analyzed.

MBC - Maximum first dimension of the array BC as given in the DIMENSION statement of the calling program.

NSU - Dimension of the continuous state-variable vector.

NCU - Dimension of the continuous control vector.

NOTE: The variables AC, MAC, BC, MBC, NSU, and NCU are only necessary for digital systems. However, the variable names must always appear in the call statement to SVANAL. MAC and MBC should be set to 1 if AC BC are dummy variables, to insure proper storage allocation.

IPO - Integer variable indicating the type of output desired.

IPO = 0 : All printout suppressed.

IPO = 1: Input data echo only. This printout is directed to logical unit number LUN1.

IPO = 2 : Printout to plotfile only. This printout is unformatted, and is directed to logical unit number LUN2. The plotfile is used for plot-only runs by SVA.

IPO = 3 : Both input data echo and plotfile information are printed out to their respective logical unit numbers.

IPO = 4 : Same as IPO=3, except input matrices are also echoed.

ISC - Integer variable indicating the type of scaling desired

ISC = 1 : unscaled singular values;

= 2 : structured singular values;

= 3 : user-input scaling matrix, D (see Section 4.2.1).

LUN1 - Logical unit number for the matrix data echo and error output. This variable should be set to the LUN of the terminal.

LUN2 - Logical unit number for the data file printout. The calling program must open and initialize the data file referenced by LUN2.

NOTE: LUN1 must be initialized to allow error statements. LUN2 need not be initialized if IPO < 2.

NWKDIM - Dimension of the work vector, WK, as given by the DIMENSION statement of the calling program. NWKDIM must be approximately

$$21 \cdot NSM^2 + 19 \cdot NC^2 + 23 \cdot M \cdot N + 31 \cdot M + 156$$

SVANAL informs you of how many elements are actually used, and gives an error message if NWKDIM is not big enough.

WK - real work vector dimensioned at least NWKDIM in the calling routine.

IWK - integer work vector dimensioned at least (2•NC + 2•M + NSM) in the calling routine.

MSV - Maximum first dimension of the array SVMAT as given in the DIMENSION statement of the calling program. MSV must be greater than NP.

C.5.3 Output Arguments

SVMAT - Two-dimensional array containing all output information. SVMAT must be dimensioned at least

[NP x (MI(1)+MI(2)+MI(3)+MI(4)+3)]

in the calling routine. Data is organized as follows:

SVMAT(I,1) : frequency, in rad/sec, I=1 to NP

SVMAT(I,2) : minimum singular value, I=1 to NP

SVMAT(I,3) : minimum eigenvalue, I=1 to NP

SVMAT(I,J+3) :
gradients with respect to the NUMPS parameters for which partial-derivative expansions were computed — J=1 to NUMPS

SVMAT(I,J+NUMPS+3) :
gradients with respect to parameters in the $\bar{A}$ matrix, — J=1 to MI(1), I=1 to NP

SVMAT(I,J+NUMPS+MI(1)+3) :
gradients with respect to parameters in the $\bar{B}$ matrix, — J=1 to MI(2), I=1 to NP

SVMAT(I,J+NUMPS+MI(1)+MI(2)+3) :
gradients with respect to parameters in the $\bar{C}$ matrix, — J=1 to MI(3), I=1 to NP

SVMAT(I,J+NUMPS+MI(1)+MI(2)+MI(3)+3) :
gradients with respect to parameters in the $\hat{P}$ matrix, — J=1 to MI(4), I=1 to NP

SVMAT(I,J+MI(1)+MI(2)+MI(3)+MI(4)+3) :
scaling (D) matrix diagonal elements, included only if ISC=1, — J=1 to NC, I=1 to NP.

C.5.4 COMMON Blocks

The COMMON blocks employed by SVANAL are:

GRADSTF, SAVE, and PARTIALS

C.5.5 Error Messages

(1) If the information in LOCS is not consistent with the matrix dimensions, the following message will be printed:

Bad matrix reference in LOCS:
Check partial derivative expansion equations

(2) If the work array is not large enough, the following message will be printed:

THE WORK ARRAY IS NOT LARGE ENOUGH IN SVCOMP
THE MAX WORKSPACE IS <Y> AND THE LAST WORK LOCATION IS <X>

<X> is the amount of additional space needed to run the program.

The above two errors are fatal; the program will abort if they are detected. the following errors will cause slight discontinuities in the plots, but otherwise will not cause any problems.

(3) If errors occur within the ORACLS routine SNVDEC, the following message will be printed:

PASS NO. <X> THERE IS AN ERROR IN CALL TO SNVDEC, IERR = <I> IN SNGVDI

IERR can be looked up in the ORACLS manual; it usually indicates a numerical convergence problem.

(4) If (Is-A) is not invertable at some $s=j\omega$, the following message will be printed:

SINGULARITY OCCURED IN ISMINA

(5) If the subroutine MAKEUV has numerical difficulties at some frequency, it will print the message

MAKEUV FAILED TO ANALYZE UV MATRIX

These numerical problems can be ignored unless they happen more than two or three times during a run.

C.5.6 Subroutines Employed by SVANAL

The following subroutines from ORACLS and FRL are employed.

FRL - SVCOMP, ISMINA, SVGRAD, DIGSVG, EXTRACT, DLESELM, MAKEUV, REALEL, BLKDIAG, PUTMAT

MODIFIED
ORACLS - PRNT1, LNCNT1, SCALEM, SNVDEC, SYSSLV, SCALEM

MODIFIED
EISPAK - ELTRAN

ORACLS - ADD, NULL, MULT, TRANP, JUXTR, JUXTC, UNITY, EQUATE, EXPINT, EIGEN, GELIM, ELMHES, BALANC, HQR, DETFAC, SHRSLV, SCHUR, INVIT, ELMBAK, BALBAK, NORMS, MAXEL, HSHLDR, BCKMLT

C.5.7 Subroutines Employing SVANAL

None

END

DATE

DEC. 1, 1987